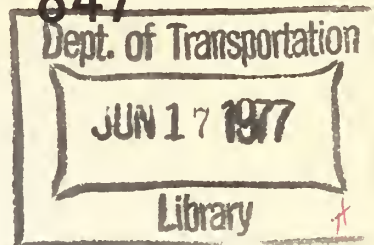


TL
243
.W36
v.2

DOT HS-801 847



FINITE ELEMENT ANALYSIS OF AUTOMOTIVE STRUCTURES UNDER CRASH LOADINGS Volume II-Technical Report

Contract No. DOT-HS-105-3-697

March 1976

Final Report

PREPARED FOR:

U.S. DEPARTMENT OF TRANSPORTATION

NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION

WASHINGTON, D.C. 20590

The contents of this report reflect the views of IIT Research Institute which is responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policy of the Department of Transportation. This report does not constitute a standard, specification or regulation.

TL
243
.W36
v.2

Dept. of Transportation

JUN 17 1977

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. DOT HS-801 847	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle FINITE ELEMENT ANALYSIS OF AUTOMOTIVE STRUCTURES UNDER CRASH LOADINGS, Volume II. Technical Report		5. Report Date March 1976	
		6. Performing Organization Code	
7. Author(s) R. E. Welch; R. W. Bruce; T. Belytschko		8. Performing Organization Report No. J6321	
9. Performing Organization Name and Address IIT Research Institute 10 West 35 Street Chicago, Illinois 60616		10. Work Unit No.	
		11. Contract or Grant No. DOT-HS-105-3-697	
12. Sponsoring Agency Name and Address Department of Transportation, NHTSA 400 7 Street S.W. Washington, D.C. 20590		13. Type of Report and Period Covered 6-30-1973 - 6-30-1975 Technical Final Report	
		14. Sponsoring Agency Code	
15. Supplementary Notes			
16. Abstract A research project was undertaken to develop a finite element computer program for use in the dynamic analysis of vehicle structures, including sheet metal, in a crash environment. This research program involved the following major tasks: <ul style="list-style-type: none"> ● A technique was developed for the finite element analysis of the dynamic response of plate beam structures involving very large displacements and rotations and elastic-plastic material behavior. The principal feature of this technique involves the decomposition of the element displacement field into rigid body components and deformation components thus allowing the use of a small deflection formulation in the analysis. ● A computer program was developed incorporating this analysis procedure for beam and plate elements and rigid links together with appropriate time integration procedures and material property descriptions. 			
17. Key Words finite element computer program, dynamic analysis, vehicles, crashworthiness		18. Distribution Statement Document is available to the public through the National Technical Information Service, Springfield, Virginia 22161	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 157	22. Price

Form DOT F 1700.7 (8-69)

16. Abstract (Concl)

- A substantial number of test and demonstration problems were analyzed with this computer program ranging from simple classical solutions for beams and plates through large scale simulations of actual crash tests.

PREFACE

This final report, entitled "Finite Element Analysis of Automotive Structures under Crash Loadings" presents the results of a research project undertaken from 1 July 1973 to 30 June 1975 by IIT Research Institute (IITRI) for the Department of Transportation, National Traffic and Safety Administration (NHTSA) under Contract DOT-HS-105-3-697 (IITRI Project J6321). The report is presented in two volumes: Volume I, Summary Final Report, and Volume II, Technical Final Report.

The NHTSA Contract Technical Manager initially was Dr. Michael Chi and subsequently Mr. James Hackney. Dr. R. E. Welch served as the IITRI Project Manager under general supervision of Mr. A. Longinow, Manager, Structural Analysis Section, and Dr. K. E. McKee, Director, Engineering Mechanics. IITRI personnel who made significant technical contributions to the project are R. W. Bruce, M. Smith and R. E. Welch. Prof. T. Belytschko of the Circle Campus, University of Illinois, served as a consultant to the project.



CONTENTS

<u>Section</u>	<u>Page</u>
1. INTRODUCTION	1
1.1 Previous Analyses	1
1.2 Analysis Requirements	3
1.2.1 Solution Procedure	4
1.2.2 Element Formulation	4
1.2.3 Material Properties	5
1.2.4 Program Validation	5
1.3 Report Organization	6
2. ANALYSIS FORMULATION	7
2.1 Equations of Motion	7
2.2 Time Integration	9
2.3 Element Coordinate Systems and Definitions	10
2.3.1 Plate Elements	11
2.3.2 Beam Elements	14
2.3.3 Rigid Links	17
2.4 Element Force	18
2.4.1 Plate Elements	19
2.4.2 Beam Element	26
2.5 Nodal Force Transformations	30
3. ANALYSIS RESULTS	33
3.1 Test Problems	33
3.1.1 Dynamic Buckling of a Spherical Shell Segment	33
3.1.2 Dynamic Response of a Cylindrical Panel	36
3.2 Barrier Test Simulations	43
3.2.1 Stub Frame Impact (Test 283-51)	47
3.2.2 Stub Frame and Sheet Metal Impact (Test 283-53)	52
4. CONCLUSIONS AND RECOMMENDATIONS	63
4.1 Conclusions	63
4.2 Recommendations	64
APPENDIX A - Triangular Plate Displacement and Curvature Functions	67
APPENDIX B - Computer Program Input	73
APPENDIX C - Program Description	81
APPENDIX D - Program Listings	89
REFERENCES	155



LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Page</u>
1. Three-Dimensional Plate Element	12
2. Three-Dimensional Beam Element	15
3. Triangular Plate Element	20
4. Uniaxial Bilinear Stress-Strain Relation	25
5. Beam Coordinates and Cross-Sectional Geometry	28
6. Spherical Shell Finite Element Mesh	37
7. Displacement-Time History of a Clamped Spherical Shell	38
8. Progressive Deformation Stages of a Clamped Spherical Shell	39
9. Cylindrical Shell - Geometry and Properties	40
10. Cylindrical Shell Segment - Undeformed	41
11. Cylindrical Shell Segment - Crown Displacement versus Time (Node 5)	42
12. Cylindrical Shell Segment - Crown Velocity versus Time	44
13. Cylindrical Shell Segment - Deformed	45
14. 30 mph Stub Frame Impact at 40 msec (Top)	48
15. 30 mph Stub Frame Impact at 40 msec (Side)	49
16. Vehicle Center of Gravity Velocity for 30 mph Stub Frame Test	53
17. Vehicle Center of Gravity Acceleration for 30 mph Stub Frame Test	54
18. Stub Frame Test - Barrier Force	55
19. Stub Frame and Sheet Metal Model (Undeformed)	56
20. Stub Frame and Sheet Metal Model (Deformed, 20 msec)	57

LIST OF ILLUSTRATIONS (Concl)

<u>Figure</u>		<u>Page</u>
21.	Stub Frame and Sheet Metal Vehicle Center of Gravity Acceleration	60
22.	Stub Frame and Sheet Metal Total Barrier Force	61
C-1	Computer Program Structure	83

LIST OF TABLES

<u>Table</u>		
1.	Summary of Test Problems	34
2.	Plymouth Fury Test Series	46
3.	Stub Frame Beam Properties	50

1. INTRODUCTION

The increased concern in recent years over vehicle safety has focused, in part, on the ability of a vehicle to sustain a crash event; to absorb, redirect and otherwise manage the severe forces and energy associated with the event; and thereby, to lessen the severity of the environment to which passengers are exposed. In addition to basic vehicle crashworthiness, serious attention has also been devoted to such factors as the propensity of a vehicle to damage in minor crash events and the threat posed to pedestrians or other vehicles by particular vehicle designs.

Since quantitative knowledge of the deformation characteristics of vehicle structures is an intrinsic requirement in all considerations of this type, it is not surprising that the developing concern for vehicle safety has given rise to intense and continuing efforts to provide mathematical descriptions of structural deformations in the crash environment. This report presents the results of a research project undertaken by IIT Research Institute (IITRI) for the Department of Transportation to develop a finite element computer program for use in the dynamic analysis of vehicle structure, including sheet metal, in a crash environment.

1.1 Previous Analyses

The literature relating to vehicle crash simulations is extensive and has been summarized by several previous investigators (Refs. 1 through 5). Present analytical procedures fall largely into two broad categories which may be termed lumped parameter or equivalent systems and formal approximation techniques (primarily finite element). In the former category, the vehicle is treated as an assemblage of lumped masses and resistance elements which are selected to represent specific subassemblies and components. The properties of the various elements of such models are determined by a static and dynamic crush testing, direct measurement, and the judicious use of simple analysis procedures. The use of such techniques has become widespread in studies of vehicle crashworthiness

as represented by Tani and Emory (Ref. 6), Kamal (Ref. 7), and the models developed by Battelle (Ref. 8) and Dynamic Sciences (Ref. 9). Such techniques are valuable in that they provide an efficient and inexpensive method of simulation, are fairly accurate for well established vehicle configurations and, in effect, serve as a depository for previous test and design results. The limitations of this type of procedure are principally that they are heavily reliant on previous testing; largely limited to well-established vehicle configurations and only weakly tied to basic principles of structural behavior.

The finite element approach represents an attempt to make up for the limitations inherent in the lumped parameter analyses, by employing more formal approximation techniques in the discretization of the structure and a greater reliance on more fundamental structural representations and properties. The limitations of this approach are found in the inherent tendency toward more complicated and expensive computations and the difficulties found in modeling the extensively complex phenomena associated with the crash environment. Such phenomena include large deflections and rotations in the deformed structure, regions of intense curvature (wrinkling), material strain rate effects and interference and contact among structural components during the response. These procedures, indeed, are not totally free of a reliance on testing and analytical judgment and, in fact, can be viewed as a somewhat more formal lumped parameter system and used in connection with the more empirical procedure in hybrid, combination models.

A variety of finite element analyses directed toward the dynamic analysis of vehicle structures have been reported previously. Selna and Salinas (Ref. 10) report a three-dimensional, linear elastic dynamic analysis of a two-door sedan in a 60 mph frontal barrier impact. Their results indicate that significant material yielding and collapse would occur and clearly demonstrate the body and engine pitch motions which ensue in such a crash. Kirioka (Refs. 11 through 13) in a series of papers have reported an elaborate computational system for the analysis of vehicle structures. The analysis includes nonlinear material and geometric effects, and

static, dynamic, and modal procedures and employs graphic systems and mesh generation as analysis aids. Thompson (Ref. 14) has investigated the side-on collision of passenger vehicles using finite element beam elements to model the vehicle structure. The elements include nonlinear material properties and a geometric stiffness matrix for beam columns and shows reasonable correlation with test data despite the lack of plate and membrane elements in the analysis. Shieh (Ref. 15) has developed a framework analysis for the precollapse dynamic response of plane, ideal, elastic-plastic frame structures which he applies to the analysis of framework devices used to attenuate crash effects. Finally, Young (Ref. 16) and Melosh (Ref. 17) have reported on an analysis employing an energy minimization technique for large displacements of elastic-plastic beam and truss elements. Results reported in Ref. 17 suggest that although the equivalent framework used is substantially stiffer than the actual structure the computed responses are similar in character to those obtained in test.

1.2 Analysis Requirements

The response of vehicle structures under crash loadings is a complex process primarily involving:

- transient, dynamic behavior
- complicated framework and shell assemblages
- large deflections and rotations
- extensive plastic deformation.

Previous attempts at a formal analysis of this process have been only partly successful due to a variety of limitations which, in particular instances, have included inadequacies in element formulations, material representations or solution procedures. The work presented in this report represents an attempt to develop a finite element program which is specifically tailored to the class of problems inherent in vehicle crash response, and which employs or extends current avenues in finite element analysis which seem best suited to such problems. The field of nonlinear finite element analysis is currently an extremely active area of research with an extensive, related literature and a variety of methods

and approaches. Consequently, a formal review of the field as background for the present analysis approach is not attempted. Instead, major features of the present work are briefly described, and some rationale is offered for their use in the context of vehicle analysis.

1.2.1 Solution Procedure - The analysis procedure is based on explicit, timewise numerical integration of the equations of motion for the node points (three translations and three rotations per node). The choice of an explicit, rather than implicit, procedure is exploited throughout the analysis by using "lumped" nodal masses and by calculating the internal forces at the nodes from direct integration of the element stress fields without reference to element or assembled stiffness matrices for the structure. The relative merits of implicit and explicit procedures and the importance of the mass formulation have been studied by Krieg and Key (Ref. 18), and the use of a direct calculation of internal forces has been described by Belytschko and Chiapetta (Ref. 19) and by Oden (Ref. 20). The choice of an explicit integration procedure and a direct calculation of nodal forces results in a program with minimum (but still substantial) computer storage requirements. This, in turn, is equivalent to a capability of processing reasonably detailed and extensive structural models with relative ease. An early capability for handling relatively large problems is considered essential in developing realistic simulations of actual crash events because of the complex geometry and construction found in real vehicles. Furthermore, this procedure lends itself to a reasonably well organized modular program which admits subsequent change and development, including the further incorporation of an implicit method.

1.2.2 Element Formulation - The treatment of large displacements and rotations employs a decomposition of the element displacement field into a rigid body rotation and translation associated with a local coordinate system attached to and moving with the element, and a remaining displacement field which describes the deformation of the element relative to the current position of the

element axes. This transformation, in effect, removes the average rigid body rotation of the element and allows the use of small or moderate deflection element formulations in the calculation of element forces. In this manner, extremely large rotations and deflections can be accommodated by the analysis with accuracy depending primarily on the size of the elements relative to the curvature of the structure. Although a formal convergence theorem is lacking, the decomposition does hold for infinitesimal regions and numerical studies show excellent agreement with classical solutions. Previous applications of this technique may be found in work by Argyris, Kelsey and Kamel (Ref. 21); Wempner (Ref. 22); Murray and Wilson (Ref. 23); Belytschko and Hsieh (Ref. 24); and other references which are cited in Section 2. The computer program at present includes low order triangular plate elements and three-dimensional beam elements. A triangular membrane-hinge line element is also available but is not presently compatible with the beam formulation.

1.2.3 Material Properties - The computer program currently uses simple elastic-plastic stress strain laws; a uniaxial relation for beam elements and a biaxial strain hardening Mises model for plates. Element forces and bending moments for given strain fields are calculated by piecewise linear numerical integration of the stresses at selected points in the cross section. The program is designed to accommodate other constitutive relations readily or to accept explicit relations among thrust, moment, extension and curvature. In fact, linear relationships are provided at this level which result in more efficient computations for this class of problems.

1.2.4 Program Validation - An extensive series of test problems were investigated throughout the program development to ensure that the basic formulation as well as the numerical procedures and programming were performing properly. These range from fairly simple cases such as beam bending, wave propagation and arch buckling problems which were designed to check the basic formulations, through more complex situations for which previously published numerical solutions could be found. Finally, a large demonstration was

attempted consisting of the simulation of actual vehicle-barrier tests. In a subsequent section, the test cases are summarized and the large-scale simulation is described in detail.

1.3 Report Organization

Section 2 of this report contains a description of the analysis formulation in fairly complete detail. In Section 3 the test problems are summarized and a detailed description of the large-scale vehicle-barrier test simulation is provided. A summary of the results of the investigation and recommendations for further application and development of the computer program are given in Section 4. Appendix A gives the complete plate element curvature functions and detailed descriptions of the program input, program operations and program listings are provided in Appendixes B, C, and D., respectively.

2. ANALYSIS FORMULATION

This description of the analysis formulation is intended to roughly parallel the sequence of computation in the computer program. A typical program cycle is: integrate equations of motion; transform displacement to element coordinate systems; calculate element forces; and transform forces to global and nodal coordinate systems. Solution procedures employed in the computer program are also described.

2.1 Equations of Motion

The equations of motion at a typical node point* for translations and rotations respectively, are written as follows:

Translations

$$[M][\ddot{u}_G] = [f_G^E] - [f_G^I] \quad (1)$$

where

$[M]$ - the diagonal, lumped mass matrix for the node

$[\ddot{u}_G] = [\ddot{u}_{xG}, \ddot{u}_{yG}, \ddot{u}_{zG}]$ is the acceleration vector for the node referred to a global system $[X_G, Y_G, Z_G]$ common to all nodes

$[f_G^E]$ - the vector of external, applied loads at the node referred to the global system

$[f_G^I]$ - the vector of forces at the node in the global system which results from deformations in all elements associated with the node.

* Unless otherwise specified, all equations and variables refer to a typical node or element, depending on context, rather than to the ensemble for all nodes or elements. Since there is no global stiffness matrix the formation of the ensembles is relatively simple.

Rotation

$$\bar{I}_x \bar{\alpha}_x = \bar{m}_x^E - \bar{m}_x^I + \bar{\omega}_y \bar{\omega}_z (\bar{I}_y - \bar{I}_z) \quad (2a)$$

$$\bar{I}_y \bar{\alpha}_y = \bar{m}_y^E - \bar{m}_y^I + \bar{\omega}_z \bar{\omega}_x (\bar{I}_z - \bar{I}_x) \quad (2b)$$

$$\bar{I}_z \bar{\alpha}_z = \bar{m}_z^E - \bar{m}_z^I + \bar{\omega}_x \bar{\omega}_y (\bar{I}_x - \bar{I}_y) \quad (2c)$$

where all components are referred to a coordinate system $[\bar{x}, \bar{y}, \bar{z}]$ which coincides with the principal axes of the lumped mass moments of inertia at each node and which rotates with the node (i.e., a set of rigid body axes for each node) and where $[\bar{I}_x, \bar{I}_y, \bar{I}_z]$ are the principal mass moments, $[\bar{\alpha}_x, \bar{\alpha}_y, \bar{\alpha}_z]$ are instantaneous angular accelerations, $[\bar{\omega}_x, \bar{\omega}_y, \bar{\omega}_z]$ are the angular velocities, and where $[\bar{m}^E]$ and $[\bar{m}^I]$ are the external and internal moment vectors at the node point.

The orientation of the nodal axes $[\bar{x}, \bar{y}, \bar{z}]$ with respect to the global axes at any time during the motion is established by the components of three unit vectors $\vec{b}_1, \vec{b}_2, \vec{b}_3$ which remain fixed along the nodal axes $[\bar{x}, \bar{y}, \bar{z}]$, respectively, as the node rotates. If the global components of these three unit vectors are

$$\vec{b}_1: (b_{x\bar{x}}, b_{y\bar{x}}, b_{z\bar{x}}) \quad (3a)$$

$$\vec{b}_2: (b_{x\bar{y}}, b_{y\bar{y}}, b_{z\bar{y}}) \quad (3b)$$

$$\vec{b}_3: (b_{x\bar{z}}, b_{y\bar{z}}, b_{z\bar{z}}) \quad (3c)$$

then the components of any vector, \vec{v} , transform from the nodal to global system by the transformation

$$[v_G] = [B][\bar{v}] \quad (4)$$

where the columns of $[B]$ are the components of the three unit vectors

$$\begin{bmatrix} b_{x\bar{x}} & b_{x\bar{y}} & b_{x\bar{z}} \\ b_{y\bar{x}} & b_{y\bar{y}} & b_{y\bar{z}} \\ b_{z\bar{x}} & b_{z\bar{y}} & b_{z\bar{z}} \end{bmatrix} \quad (5)$$

Initially, the elements of the $[B]$ matrix are obtained from the orientation of the principal axes of the mass moments of inertia for the node in the undeformed structure and are subsequently calculated from the integration of the rotational equations of motion; equations (2).

2.2 Time Integration

The numerical technique employed to integrate the equations of motion consists of the Newmark-beta method (Ref. 25). This method relates displacement, velocity, and acceleration at the beginning (u_0, v_0, a_0) and end (u_1, v_1, a_1) of a time interval, h , by the relations

$$u_1 = u_0 + v_0 h + \left(\frac{1}{2} - \beta\right) a_0 h^2 + \beta h^2 a_1 \quad (6a)$$

$$v_1 = v_0 + \frac{h}{2} (a_0 + a_1) \quad (6b)$$

where β is an assumed parameter related to an assumed variation of acceleration with time over the interval. Solutions to the equations of motion, equations (1) and (2), may be obtained for a proper choice of β by using equations (6), (1), and (2) in an iterative manner, predicting u_1 and v_1 by means of equations (6) and correcting a_1 by means of equations (1) and (2) until the process converges. The full solution procedure, with iteration, is available in the computer program but early trials indicated that the most efficient process was obtained with $\beta = 0$ and no iterations, and all subsequent computations have been carried out with these parameters.

The solution procedure is applied to the equations of motion for translational degrees of freedom, equation (1), exactly in the manner outlined above. The treatment of the rotational degrees of freedom, equations (2), differs slightly in that the rotation term (analogous to u_0 or u_1) has no physical significance as calculated and is not subsequently used in the calculations. Instead of an explicit calculation of nodal displacements or Euler angles, an auxiliary numerical integration is employed to calculate the current value of the [B] transformation based on its previous value and the current values of angular velocity and acceleration. Applying the integration algorithm (with $\beta = 0$) to a typical unit vector, \vec{b}_3 , results in

$$\vec{b}_3|_{t+h} = \vec{b}_3|_t + \dot{\vec{b}}_3 h + \frac{1}{2} h^2 \ddot{\vec{b}}_3 \quad (7)$$

The derivatives of the unit vector are related to the angular velocity and acceleration by the relations

$$\dot{\vec{b}}_3 = \vec{\omega} \times \vec{b}_3 \quad (8a)$$

$$\begin{aligned} \ddot{\vec{b}}_3 &= \dot{\vec{\omega}} \times \vec{b}_3 + \vec{\omega} \times \dot{\vec{b}}_3 \\ &= \dot{\vec{\omega}} \times \vec{b}_3 + \vec{\omega} \times \vec{\omega} \times \vec{b}_3 \end{aligned} \quad (8b)$$

Expanding into scalar components in the previous $[\bar{x}, \bar{y}, \bar{z}]$ system, these relations become

$$b_{3\bar{x}}|_{t+h} = + \bar{\omega}_y h + \frac{h^2}{2} (\bar{\omega}_x \bar{\omega}_z + \alpha_y) \quad (9a)$$

$$b_{3\bar{y}}|_{t+h} = - \bar{\omega}_x h + \frac{h^2}{2} (\bar{\omega}_y \bar{\omega}_z + \alpha_x) \quad (9b)$$

with the third component, $b_{3\bar{z}}$, obtained from the condition that \vec{b}_3 remains a unit vector. A similar process is applied to another of the unit vectors, say \vec{b}_2 , and the third vector is obtained from the cross product relation, $\vec{b}_1 = \vec{b}_2 \times \vec{b}_3$. The components of the new unit vectors referred to the global coordinate system then constitute the current transformation matrix $[B]$.

2.3 Element Coordinate Systems and Deformations

A coordinate system $[\hat{x}, \hat{y}, \hat{z}]$ is embedded in each element and serves to define the rigid body rotation of the element and as a reference for element distortions and forces. The components of a vector, \vec{V} , transform from the element coordinate system to the global coordinate system by the time varying transformation

$$[v_G] = [E][\hat{v}] \quad (10)$$

where the elements of E are

$$\begin{bmatrix} e_{x\hat{x}} & e_{x\hat{y}} & e_{x\hat{z}} \\ e_{y\hat{x}} & e_{y\hat{y}} & e_{y\hat{z}} \\ e_{z\hat{x}} & e_{z\hat{y}} & e_{z\hat{z}} \end{bmatrix} \quad (11)$$

which correspond columnwise with the global components of unit vectors $\vec{e}_1, \vec{e}_2, \vec{e}_3$ oriented with the element axes $[\hat{x}, \hat{y}, \hat{z}]$ respectively, that is

$$\vec{e}_1: [e_{xx}, e_{yx}, e_{zx}] \quad (12a)$$

$$\vec{e}_2: [e_{xy}, e_{yy}, e_{zy}] \quad (12b)$$

$$\vec{e}_3: [e_{xz}, e_{yz}, e_{zz}] \quad (12c)$$

2.3.1 Plate Elements - The element coordinate system for the three-dimensional plate element (Figure 1) is constructed from the current positions (i.e., undeformed position plus displacement) of the three points (I,J,K) which define the element. In terms of unit vectors pointing from I to J (\vec{S}_{IJ}) and from I to K (\vec{S}_{IK}) the global components of the three unit vectors, $\vec{e}_1, \vec{e}_2, \vec{e}_3$ are calculated as

$$\vec{e}_1 = (\vec{S}_{IJ} + \vec{S}_{IK})/2 \quad (13a)$$

$$\vec{e}_3 = \vec{S}_{IJ} \times \vec{S}_{IK} \quad (13b)$$

$$\vec{e}_2 = \vec{e}_3 \times \vec{e}_1 \quad (13c)$$

These operations apply equally well to both the deformed and undeformed positions of the element and, in particular, define \vec{e}_3^0 , the normal to the undeformed plate surface, which is subsequently employed to establish element rotations in the deformed position.

Element displacements and rotations are defined with respect to the element coordinate system and are determined from the global components of displacements and the transformations [B] associated with the three node points I,J,K. Having determined the [E] transformation as described above the global components of displacement, u_G , at each of the nodes as determined from the integration of equation (1) can be transformed to the element system by the operation

$$[\hat{u}] = [E]^T [u_G] \quad (14)$$

where $[E]^T$ denotes the transpose of [E]. Since the three node points uniquely determine the transformation [E] this calculation

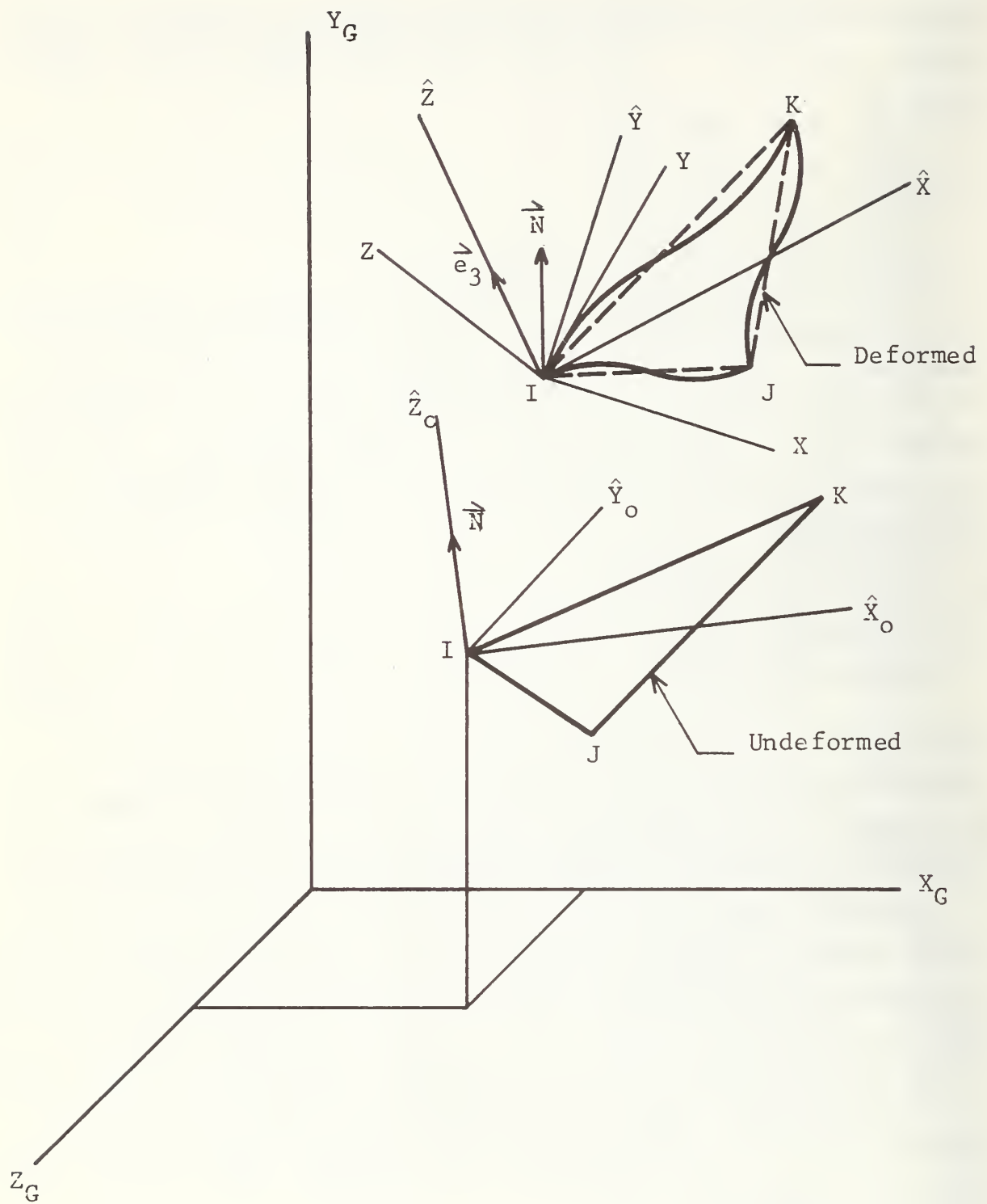


Figure 1 Three-Dimensional Plate Element

would eliminate the component of displacement normal to the plane of the plate, $\hat{u}_z = 0$. For a small deflection plate formulation then the nodal displacements \hat{u}_x, \hat{u}_y lie in the plane IJK and determine the membrane midsurface strains while the plate curvatures are determined by the rotations alone. The computer program actually employs a somewhat different, but equivalent and more efficient, approach to the calculation of membrane strains. The lengths of the element sides in the deformed position are calculated directly from the global displacements and these new lengths determine the extensions parallel to the three element sides. For instance, side IJ

$$E_{IJ} = \frac{L_{IJ} - L_{IJ}^0}{L_{IJ}^0} \quad (15)$$

where L_{IJ} and L_{IJ}^0 are the lengths of the side IJ in the deformed and undeformed positions respectively. Since a constant membrane strain element is employed the three extensions uniquely determine the membrane strain in the element.

The calculation of rotations at a typical node for a plate element involves both the [E] transformation for the element and the [B] transformation for the node and is based on the usual thin plate assumption that lines which are initially straight and normal to the surface of the plate remain so during the deformation. In the present terminology, an initial normal to the plate, \vec{e}_3^0 , located at node I rotates with the nodal axes (i.e., $[\bar{x}, \bar{y}, \bar{z}]$ system) at node I and its components in the nodal system remain unchanged during the motion. These components are given by

$$[\bar{n}] = [B_0]^T [E_0] [\hat{e}_3^0] \quad (16)$$

where

$$[\hat{e}_3^0] = [0, 0, 1]^T$$

and $[\bar{n}]$ are the components of the normal to the deformed surface, \vec{n} , at node I. It should be noted that since the components of \vec{n} are constant in the nodal coordinate system which rotates during

the deformation, the components of \vec{n} in the global or element systems are time varying. At any time during the deformation the angle, θ , between the normal to the deformed plate at node I, \vec{n} , and the current normal to the rigid body plane of the plate, \vec{e}_3 , may be calculated from the vector cross product

$$\vec{\theta} = \sin\theta \vec{e}_\theta = \vec{e}_3 \times \vec{n} \quad (17)$$

Assuming that this is a sufficiently small angle, its components in the element system are

$$\hat{\theta}_x = -\hat{n}_y \quad (18a)$$

$$\hat{\theta}_y = \hat{n}_x \quad (18b)$$

where the components of the deformed normal, \vec{n} , in the element system are determined from

$$[\hat{n}] = [E]^T [B] [\bar{n}] \quad (19)$$

The rotations $\hat{\theta}_x, \hat{\theta}_y$ (two per node per element) are used in subsequent calculations of element curvatures and, in turn, element forces and moments

2.3.2 Beam Elements - The element coordinate system for the three-dimensional beam element (Figure 2) is constructed somewhat differently than for the plate element. In the undeformed position, the unit vector \vec{e}_1^0 lies along the axis of the beam (the \hat{x}_0 axis) and is defined by the positions of the end point I and J. The unit vector, \vec{e}_2^0 (the \hat{y}_0 axis) is directed normal to the beam axis and lying in a plane defined by a reference or pointer node specified in the input data and containing one of the principal axes of the beam cross section. The third unit vector is determined by $\vec{e}_3^0 = \vec{e}_1^0 \times \vec{e}_2^0$ and establishes the \hat{z}_0 axis. In this case two unit vectors $\vec{\eta}$ and $\vec{\xi}$ which initially coincide with the undeformed \hat{x}_0 and \hat{y}_0 axes are embedded in the nodal coordinate system $[\bar{x}, \bar{y}, \bar{z}]$ at each end of the beam. The nodal components of these vectors are

$$[\bar{\eta}] = [B_0]^T [E_0] [\hat{e}_1^0] \quad (20)$$

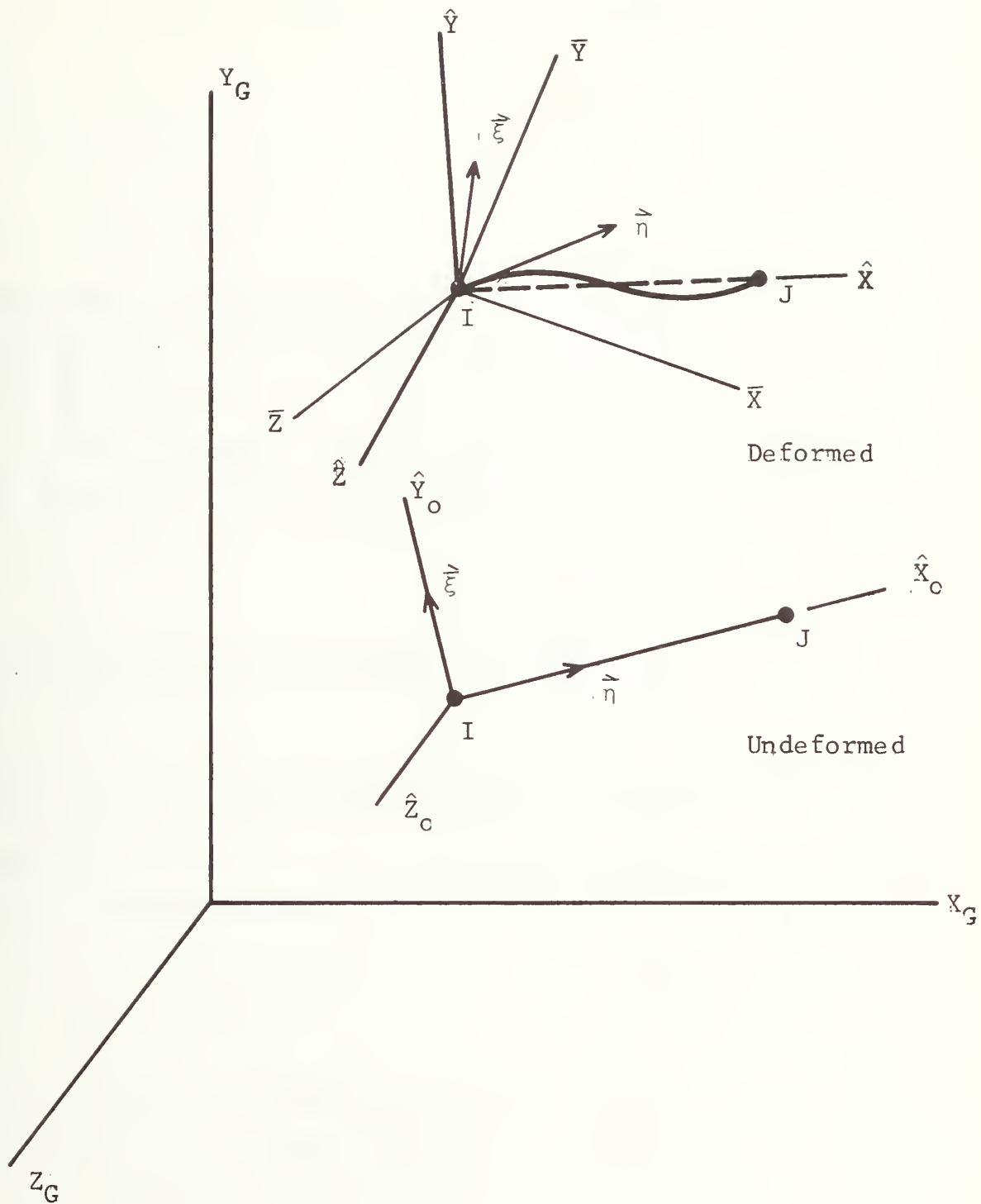


Figure 2 Three-Dimensional Beam Element

where

$$[\hat{e}_1^0] = [1, 0, 0]^T$$

and

$$[\bar{\xi}] = [B_0]^T [E_0] [\hat{e}_2^0] \quad (21)$$

where

$$[\hat{e}_2^0] = [0, 1, 0]$$

The construction of element axes for the deformed beam differs somewhat from the procedure employed for the plate since the position data for the two nodes I and J are not sufficient to locate the \hat{y} and \hat{z} axes uniquely, although they are sufficient to establish the \hat{x} axes and the \vec{e}_1 unit vector directly. The \hat{z} is taken to be normal to the plane defined by the \hat{x} axis and the average of the unit vectors $\vec{\xi}_I$ and $\vec{\xi}_J$ at the ends of the beam

$$\vec{e}_3 = \vec{e}_1 \times (\vec{\xi}_I + \vec{\xi}_J) / 2 \quad (22)$$

and the \hat{y} axis is formed to complete the right-handed system

$$\vec{e}_2 = \vec{e}_3 \times \vec{e}_1 \quad (23)$$

Displacements in the element coordinate system are determined in a manner analogous to that used for the plate element. The extension of the beam is calculated directly from the deformed and undeformed lengths as determined by the displacement data for the end points

$$E_{IJ} = \frac{L_{IJ} - L_{IJ}^0}{L_{IJ}^0} \quad (24)$$

The rotations related to bending at node I are found from the unit vector $\vec{\eta}$ which defines the deformed position of the beam axis at the node and \vec{e}_1 which lies along the \hat{x} axis by means of the relation

$$\vec{\theta} = \vec{e}_1 \times \vec{\eta} \quad (25)$$

For small angles this results in

$$\hat{\theta}_z = \hat{\eta}_y \quad (26a)$$

$$\hat{\theta}_y = -\hat{\eta}_z \quad (26b)$$

where the components of $\vec{\eta}$ in the element system are determined from

$$[\hat{\eta}] = [E]^T [B] [\bar{\eta}] \quad (27)$$

The torsional rotation $\hat{\theta}$ is determined directly from the cross product relation $(\vec{\xi}_I \times \vec{\xi}_J)$ in terms of components $[\hat{\eta}_I]$, $[\hat{\eta}_J]$ in the element system.

2.3.3 Rigid Links – The computer program incorporates a rigid link or master-slave node relationship whereby one or more nodes may be identified as being rigidly connected to and driven by another master node. The slave or driven nodes have no mass properties and their motions are determined by the displacements and rotations of the master node and the appropriate kinematic relations for a connecting rigid link which is embedded in the nodal (rigid body) axes for the master. This feature is intended to be used as a means of introducing rigid body masses into a structural model to simulate such items as a vehicle center of gravity, engine blocks, wheel suspensions, etc.

In terms of the nodal axes $[\bar{x}, \bar{y}, \bar{z}]$ and transformations $[B]$ previously defined, the equations governing these links can be developed in a straightforward manner. If I and J are taken to denote the slave and master nodes, respectively, then the position of the slave node with respect to the master in the undeformed structure may be written as

$$[R_{IG}] = [R_{JG}] + [\Delta_G^0] \quad (28)$$

where $[R_{IG}] = [X_{IG}, Y_{IG}, Z_{IG}]$ is the position vector for node I in global coordinates and $[\Delta_G^0]$ is a vector whose components are the initial differences in position between the slave and master nodes. A rigid body motion of node I with respect to node J implies the components of $[\Delta_G^0]$ in the nodal coordinate system at node J are constant during the motion.

These components are

$$[\bar{\Delta}] = [B_{Jo}]^T [\Delta_G^0] \quad (29)$$

Thus, at any time during the motion the position of node I is determined from the position of node J by

$$[R_{IG}] + [u_{IG}] = [R_{JG}] + [u_{JG}] + [\Delta_G] \quad (30)$$

where $[u_{IG}]$ and $[u_{JG}]$ are the displacements of nodes I and J and $[\Delta_G]$ are the current global components of the rigid link as given by

$$[\Delta_G] = [B_J][\bar{\Delta}] \quad (31)$$

Equation (30) may be rearranged to give the displacement of node I directly as

$$[u_{IG}] = [u_{JG}] + [\Delta_G] - [\Delta_G^0] \quad (32)$$

2.4 Element Forces

Having reduced the displacements to rigid body and deformation components, the next step in the process is to calculate the forces on the nodes, taken element by element, from the displacements referred to the element coordinate systems. In general, the procedure by which the nodal forces are calculated is derived from the principal of virtual work which states that for a kinematically admissible variation in the element displacements, the work done by the nodal forces is equal to the work done by the stresses in the element. In the present notation this relation for a typical element is written as

$$[f]^T [\delta u] = \int_V [\sigma(\epsilon)]^T [\delta \epsilon] dV \quad (33)$$

where $[f]$ is the vector^{*} of all forces and moments at the nodes $(i,j,...)$ defining the element

$$[f]^T = [f_i, f_j, \dots]$$

$$[f_i]^T = [f_{ix}, f_{iy}, f_{iz}, m_{ix}, m_{iy}, m_{iz}]$$

$[u]$ is the corresponding vector of element displacements, $[\sigma]$ and $[\epsilon]$ are the biaxial stresses and strains at a point in the plate

$$[\sigma] = [\sigma_{xx}, \sigma_{yy}, \sigma_{xy}]$$

$$[\epsilon] = [\epsilon_{xx}, \epsilon_{yy}, \epsilon_{xy}]$$

and δ denotes a variation.

The strains are related to the displacements through a matrix consisting of the appropriate derivatives of the element displacement functions

$$[\epsilon] = [\Phi(x,y,z)][u] \quad (34)$$

Introducing this relation into equation (33) and noting that the resulting expressions must be satisfied term by term in the virtual displacements, leads to the following definition of the forces acting on the nodes.

$$[f]^T = \int_V [\sigma(\epsilon)]^T [\Phi] dV \quad (35)$$

The development of the nodal forces for particular elements is described below in terms that are convenient to each element formulation.

2.4.1 Plate Element — The plate element used in the program is a flat triangular element lying in the x-y plane of the element coordinate system defined by the deformed positions of the corner nodes, i,j,k (see Figure 3).

*The superscript notation $[\hat{f}], [\hat{u}]$, etc. is omitted since all quantities involved in this section are referred to a typical element coordinate system $[\hat{x}, \hat{y}, \hat{z}]$.

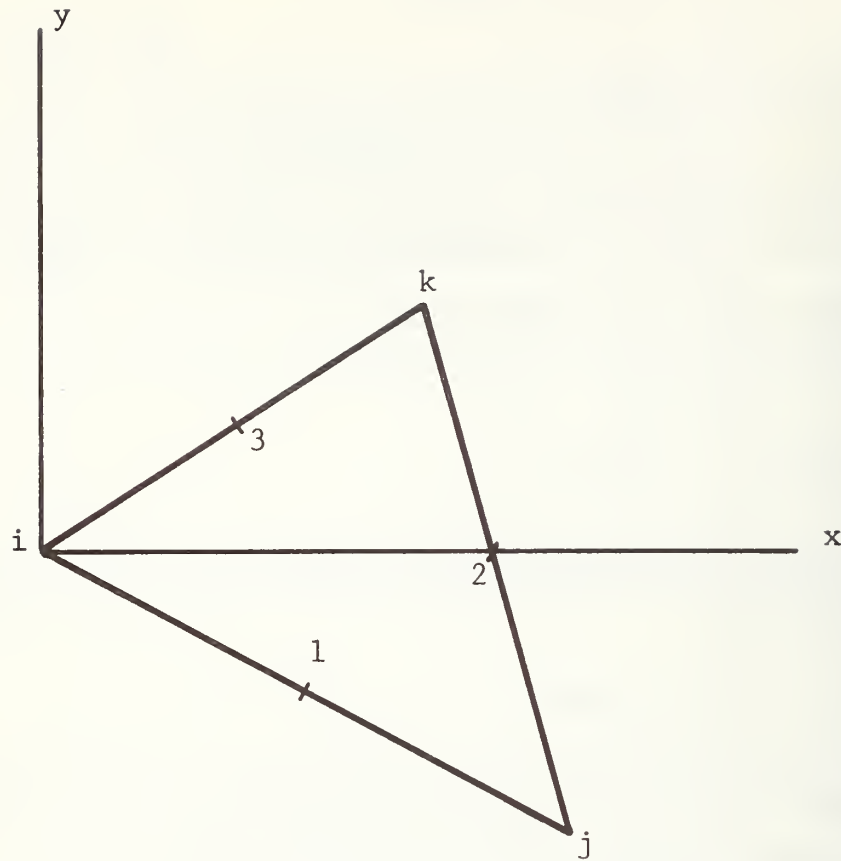


Figure 3 Triangular Plate Element

The element formulation is based on small deflection linear plate theory involving a linear displacement (i.e., constant strain) field for the midplane deformations and a cubic displacement field for the plate bending deformations as given by Zienkiewicz (Ref. 26). Node point degrees of freedom consist of the two in-plane motions u_x and u_y which determine the midplane strains and the two nodal rotations θ_x and θ_y which determine the plate curvatures. The out-of-plane displacements, u_z , of the nodes are zero since the x-y element coordinate plane was established by the deformed position of the nodes. In addition, nodal rotations normal to the plane of the plate, θ_z , are not admitted as element degrees of freedom (as in Ref. 26) for the element formulation which is employed.

Strain Displacement Relations: It is convenient to represent the element strains $[\epsilon(x,y,z)]$ in terms of the midplane strains $[\epsilon^0(x,y)]$ and curvatures $[\kappa(x,y)]$ in the form

$$[\epsilon] = [\epsilon^0] - z[\kappa] \quad (36)$$

where

$$[\epsilon^0] = [\epsilon_{xx}^0, \epsilon_{yy}^0, \epsilon_{xy}^0]^T$$

$$[\kappa] = [\kappa_{xx}, \kappa_{yy}, \kappa_{xy}]^T$$

since these quantities uncouple the midplane strain and out-of-plane bending displacement fields. By regrouping the displacements and rotations at the element nodes in a corresponding manner the midplane strains and curvatures may be written as

$$\begin{aligned} [\epsilon^0] &= [\phi][u] \\ [\kappa] &= [\psi][\theta] \end{aligned} \quad (37)$$

where $[\phi]$ and $[\psi]$ together are equivalent to $[\Phi]$ in equation (35) and

$$[u] = [u_{ix}, u_{iy}, u_{jx}, u_{jy}, u_{kx}, u_{ky}]^T$$

$$[\theta] = [\theta_{ix}, \theta_{iy}, \theta_{jx}, \theta_{jy}, \theta_{kx}, \theta_{ky}]^T$$

The elements of $[\phi]$ and $[\psi]$ are derivable from the shape functions given in Ref. 26. In particular $[\phi]$ is constant over the element and in terms of local element coordinates (with $x_i = 0$ and $y_i = 0$), has the form

$$[\phi] = \frac{1}{2A} \begin{bmatrix} y_j - y_k, & 0, & y_k, & 0, & -y_j, & 0 \\ 0, & x_k - x_j, & 0, & -x_k, & 0, & x_j \\ x_k - x_j, & y_j - y_k, & -x_k, & y_k, & x_j, & -y_j \end{bmatrix} \quad (38)$$

where A is the area of the element. The elements of $[\psi]$ vary linearly over the plate element and are obtained by differentiation of the cubic shape functions as given in Ref. 26 and by Meek

(Ref. 27) with the curvatures defined as

$$\begin{bmatrix} \kappa_{xx} \\ \kappa_{yy} \\ \kappa_{xy} \end{bmatrix} = \begin{bmatrix} \frac{\partial^2 u_z}{\partial x^2} \\ \frac{\partial^2 u_z}{\partial y^2} \\ 2 \frac{\partial^2 u_z}{\partial x \partial y} \end{bmatrix} = [\psi][\theta] \quad (39)$$

where u_z is the transverse plate displacement. The derivation of the elements of $[\psi]$ is given in detail in Appendix A.

Nodal Forces and Moments: Forces and moments acting at the nodes are calculated from equation (35) as specialized to the nomenclature for this element. The in-plane forces are obtained from the integrals

$$\begin{bmatrix} f_{ix} \\ f_{iy} \\ f_{jx} \\ f_{jy} \\ f_{kx} \\ f_{ky} \end{bmatrix}^T = \int_A [P]^T [\phi] dA \quad (40)$$

where the cross section plate forces $[P]$ are obtained by integrating through the thickness

$$[P] = \begin{bmatrix} P_{xx} \\ P_{yy} \\ P_{xy} \end{bmatrix} = \int_{-h/2}^{h/2} [\sigma(\epsilon)] dz \quad (41)$$

In a similar manner the nodal moments are calculated from integrals of the form

$$\begin{bmatrix} m_{ix} \\ m_{iy} \\ m_{jx} \\ m_{jy} \\ m_{kx} \\ m_{ky} \end{bmatrix}^T = \int_A [M]^T [\psi] dA \quad (42)$$

where the cross section plate moments $[M]$ are also obtained by integration through the thickness

$$[M] = \begin{bmatrix} M_{xx} \\ M_{yy} \\ M_{xy} \end{bmatrix} = - \int_{-h/2}^{h/2} Z [\sigma(\epsilon)] dA \quad (43)$$

The remaining nodal forces f_{iz}, f_{jz}, f_{kz} are found from overall equilibrium of the element after the nodal moments have been established. Taking moments about node i , the origin of the element coordinate system, results in the following relation for the forces

$$\begin{bmatrix} f_{jz} \\ f_{kz} \end{bmatrix} = \frac{1}{2A} \begin{bmatrix} y_3 & -x_3 \\ -y_2 & x_2 \end{bmatrix} \begin{bmatrix} m_{iy} + m_{jy} + m_{ky} \\ -m_{ix} + m_{jx} + m_{kx} \end{bmatrix} \quad (44)$$

The remaining force f_{iz} is found from transverse equilibrium

$$f_{iz} = -f_{jz} - f_{kz} \quad (45)$$

Numerical Integrations: The integrals involved in the definitions of cross section forces and moments (equations (41) and (43)) are evaluated numerically at the midpoints of the three sides of the triangle (points 1, 2, and 3 in Figure 3). The integration procedure involves a piecewise linear trapezoidal integration rule employing stresses at 'n' evenly spaced points

through the thickness with the first and last points lying on the outer surfaces of the plate. At least two points must be used since this number provides exact results for elastic problems and four or five points have proven to provide satisfactory accuracy for all cases in which detailed accuracy studies have been made. The general forms given in equations (41) and (43) apply to any consistent, nonlinear biaxial stress strain relations although applications to date have been limited to bilinear elastic-plastic material properties. It should be noted that the formal definition of internal forces and moments has been maintained in the computer program so that explicit relations among extension, curvature, force and moment may be incorporated with relative ease. This has been found useful in purely elastic problems as a means of improving the efficiency of the program.

Integrations over the surface of the plate (equations (40) and (42)) are evaluated by three point Gaussian integration using the three midside points described above. In this case (see Ref. 26) the weighting factors are 1/3 and a typical integration (equation (42) for instance) results in

$$\int_A [M]^T [\psi] dA = \frac{A}{3} \sum_L [M_L]^T [\psi_L] \quad L = 1, 2, 3 \quad (46)$$

where $[M_L]$ and $[\psi_L]$ are the values at the midside points.

Stress-Strain Relations: The stress-strain relations employed in the plate element formulation are those of a bilinear, elastic-plastic strain hardening material as indicated in Figure 4. The plastic flow calculations are based on a Mises yield criterion and isotropic hardening for the plane stress conditions which are assumed to apply in the plates. The computational algorithm follows that presented by Hartzman and Hutchinson (Ref. 28) as specialized for small strain and plane stress conditions. Given a prior stress state at a point in the plate, $\sigma_{xx}^0, \sigma_{yy}^0, \sigma_{xy}^0$ and a small increment in strain from the prior to the current state $\Delta \epsilon_{xx}, \Delta \epsilon_{yy}, \Delta \epsilon_{xy}$ the current stress state is established by the following procedure.

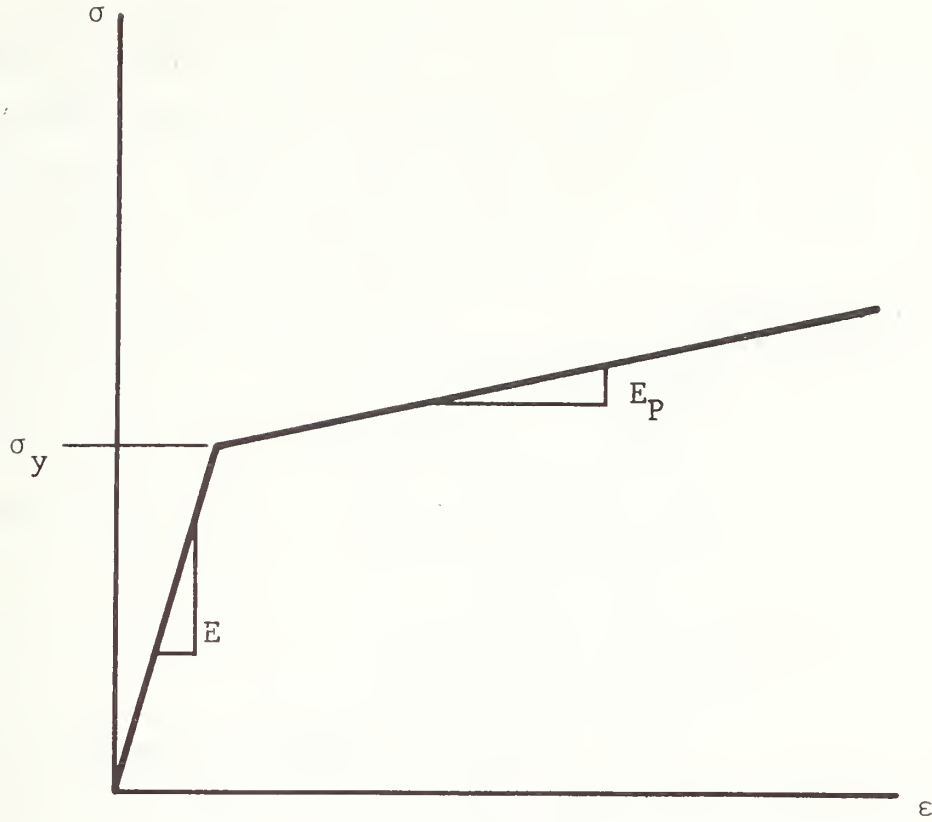


Figure 4 Uniaxial Bilinear Stress-Strain Relation

A tentative, current stress state $\overline{\sigma_{xx}}, \overline{\sigma_{yy}}, \overline{\sigma_{xy}}$ is calculated as though the strain increment were completely elastic:

$$\overline{\sigma_{xx}} = \sigma_{xx}^0 + \frac{E}{1-\nu^2} (\Delta\epsilon_{xx} + \nu \Delta\epsilon_{yy}) \quad (47)$$

$$\overline{\sigma_{yy}} = \sigma_{yy}^0 + \frac{E}{1-\nu^2} (\Delta\epsilon_{yy} + \nu \Delta\epsilon_{xx}) \quad (47)$$

$$\overline{\sigma_{xy}} = \sigma_{xy}^0 + G \Delta\epsilon_{xy}$$

where E , ν and G are the usual elastic constants. The corresponding value of effective stress for the Mises yield criterion is also calculated from the expression

$$\overline{\sigma}_e = [\overline{\sigma}_{xx}^2 - \overline{\sigma}_{xx}\overline{\sigma}_{yy} + \overline{\sigma}_{yy}^2 + \frac{3}{2}\overline{\sigma}_{xy}^2]^{1/2} \quad (48)$$

which is used to determine if plastic flow has taken place during the strain increment. If $\bar{\sigma}_e$ is less than the most recent effective stress σ_e^0 at which yielding occurred the material is unloading and the tentative stresses are, in fact, the current stress values $\sigma_{xx}, \sigma_{yy}, \sigma_{xy}$. Otherwise the tentative stress state must be modified to correct for plastic behavior. As shown in Ref. 28, the true value of the current effective stress will be

$$\sigma_e = \frac{\left[\sigma_e^0 + \left(\frac{H'}{3G} \right) \bar{\sigma}_e \right]}{1 + \left(\frac{H'}{3G} \right)} \quad (49)$$

where H' is the slope of the stress versus plastic strain curve given by

$$H' = \frac{E E_p}{E + E_p}$$

with E_p being the plastic modulus shown in Figure 4. The current stress state is then given by

$$\begin{aligned} \sigma_{xx} &= \frac{1}{(1+3\lambda)} [\bar{\sigma}_{xx} + \lambda(\bar{\sigma}_{xx} + \bar{\sigma}_{yy})] \\ \sigma_{yy} &= \frac{1}{(1+3\lambda)} [\bar{\sigma}_{yy} + \lambda(\bar{\sigma}_{xx} + \bar{\sigma}_{yy})] \\ \sigma_{xy} &= \frac{1}{1+3\lambda} [\bar{\sigma}_{xy}] \end{aligned} \quad (50)$$

where the factor, λ , is given by

$$\lambda = \frac{1}{3} \left(\frac{\bar{\sigma}_e}{\sigma_e} - 1 \right)$$

Further details concerning the development of this algorithm may be found in Ref. 28.

2.4.2 Beam Element—The beam element employed in the program is based on the conventional small deflection formulation involving cubic displacement fields (i.e., linear curvature distribution) for the transverse displacements and linear displacement distributions for the axial and torsional motions (i.e., constant axial strain and twist). The development of the element forces parallels that used for the plate with such changes and simplifications as are appropriate for beam theory.

Strain-Displacement Relations: The longitudinal strain in the beam, $\epsilon_{xx}(x)$, is written in terms of the extension at the centroid $\epsilon^0(x)$ and curvatures $\kappa_{yy}(x)$ and $\kappa_{zz}(x)$ about the principal axes of the cross section (see Figure 5) in the form

$$\epsilon_x = \epsilon^0 - y \kappa_{zz} - z \kappa_{yy} \quad (51)$$

The extensions and curvatures are in turn written in terms of the nodal displacements using the appropriate derivatives of the displacement fields which results in

$$\begin{aligned} \epsilon^0 &= \frac{(u_{xj} - u_{xi})}{L} \\ \kappa_{yy} &= 2(2-3\xi) \frac{\theta_{yi}}{L} + 2(1-3\xi) \frac{\theta_{yj}}{L} \\ \kappa_{zz} &= 2(3\xi-2) \frac{\theta_{zi}}{L} + 2(3\xi-1) \frac{\theta_{zj}}{L} \end{aligned} \quad (52)$$

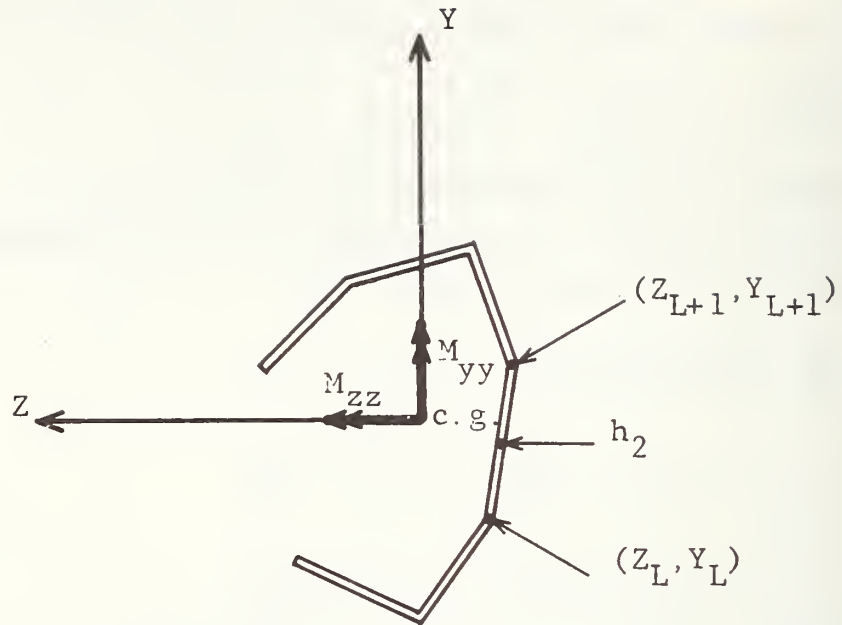
where L is the length of the beam and $\xi = X/L$.

Nodal Forces and Moments: Forces and moments acting at the nodes are determined by introducing these strain displacement relations, equations (51) and (52), into the appropriate form of equation (35) with the following results:

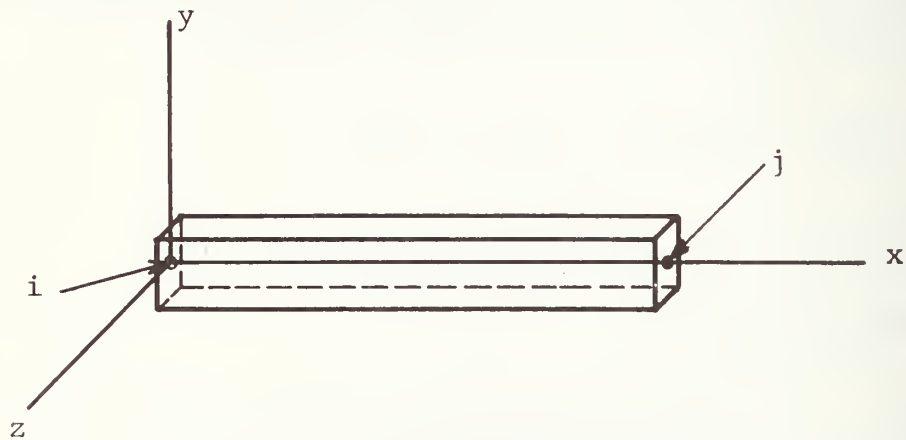
$$f_{xj} = \frac{1}{L} \int_L P_{xx} dx; \quad f_{xi} = -f_{xj} \quad (53)$$

where P_{xx} is the internal cross section force

$$\begin{aligned} P_{xx} &= \int_A \sigma_{xx} dA \\ m_{yi} &= -\frac{2}{L} \int_L (2-3\xi) M_{yy} dx \\ m_{yj} &= -\frac{2}{L} \int_L (1-3\xi) M_{yy} dx \end{aligned} \quad (54)$$



a. Cross Section Geometry and Moments (Positive Face)



b. Element Coordinate System

Figure 5 Beam Coordinates and Cross Section Geometry

where M_{yy} is the internal cross section moment* about the y axis

$$\begin{aligned} M_{yy} &= \int_A z \sigma_{xx} dx \\ m_{zi} &= \frac{2}{L} \int_L (3\xi-2) M_{zz} dx \\ m_{zy} &= \frac{2}{L} \int_L (3\xi-1) M_{zz} dx \end{aligned} \quad (55)$$

where M_{zz} is the internal cross section moment about the z axis

$$M_{zz} = - \int_A y \sigma_{xx} dA$$

The torsional moments are omitted from the formal development given above and are obtained directly in terms of the torsional stiffness for the beam section as

$$m_{xj} = K_t G \frac{(\theta_{xj} - \theta_{xi})}{L} ; \quad m_{xi} = -m_{xj} \quad (56)$$

where K_t is the torsional constant for the cross section and G is the shear modulus. Finally, the transverse nodal forces are determined from equilibrium conditions using the nodal moments as

$$\begin{aligned} f_{zj} &= \frac{m_{yi} + m_{yj}}{L} ; & f_{zi} &= -f_{zj} \\ f_{yj} &= \frac{m_{zi} + m_{zj}}{L} ; & f_{yi} &= -f_{yj} \end{aligned} \quad (57)$$

Numerical Integrations: The integrals involved in the definition of internal cross section moments and forces are obtained by numerical integration of the stress distribution over the cross section. The cross sections are treated as thin-walled plate work sections defined as indicated in Figure 5 by tables of the coordinates of points on the centerline of the wall and the thicknesses

*The definitions of the cross section moments, M_{yy} and M_{zz} are chosen to be in the positive sense on the positive face of the beam cross section and to result in the usual sign convention for elastic material, i.e., $M_{zz} = E I_{zz} \kappa_{zz}$ and $M_{yy} = -E I_{yy} \kappa_{yy}$.

of wall sections. The integration algorithm is a generalization to two dimensions (the y-z plane) of the procedure employed in the plate element and assumes a piecewise linear variation in stress around the periphery of the cross section. As in the plate cross section integrations this procedure preserves the exact elastic stiffness of the cross section provided that integration points include all corners in the cross section. The stress strain algorithm relation employed in these integrations is the one-dimensional version of that prescribed in connection with the plate element which can be obtained from equations (47) through (50) simply by deleting all occurrences of σ_{yy} and σ_{xy} . As with the plate the definitions of internal cross section forces and moments have been maintained in the computer program to facilitate subsequent use of explicit definitions of these quantities.

The integrals over the length of the beam are not carried out explicitly but instead are evaluated in terms of P_{xx} , M_{yy} , M_{zz} at the end points of the beam:

$$\begin{aligned} f_{xj} &= \frac{1}{2} \left[P_{xx} \Big|_{x=0} + P_{xx} \Big|_{x=L} \right] \\ m_{yi} &= -M_{yy} \Big|_{x=0} & m_{yj} &= M_{yy} \Big|_{x=L} \\ m_{zi} &= -M_{zz} \Big|_{x=0} & m_{zj} &= M_{zz} \Big|_{x=L} \end{aligned} \quad (58)$$

For the moments this is equivalent to assuming that the internal moments M_{yy} and M_{zz} vary linearly along the beam and is consistent with the usual beam formulation which implies constant shear and linear moment variation along the beam

2.5 Nodal Force Transformations

The preceding section describes the manner in which forces and moments acting on the nodes of a given element are calculated for specified values of nodal displacement. These are calculated on an element by element basis with reference to the individual element coordinate system $(\hat{x}, \hat{y}, \hat{z})$ and must be transformed to the global (x_G, y_G, z_G) and nodal $(\bar{x}, \bar{y}, \bar{z})$ systems respectively in order

to be entered into the appropriate equations of motion (equations (1) and (2)). For a particular element the nodal forces, $[\hat{f}]$, are transformed to the global system by means of the element transformation

$$[f_G] = [E][\hat{f}] \quad (59)$$

and entered directly into the force summations in equation (1). The moments $[\hat{m}]$ are first transformed to the global system and, if the nodes in question are not part of a master-slave relation, subsequently to the nodal coordinate system for the particular node

$$[\bar{m}] = [B]^T[E][\hat{m}] \quad (60)$$

and entered into the moment summations (equations (2)). If the node in question is associated with a master node, the moments are first transformed to the global system by the transformation

$$[m_{GI}] = [E][\hat{m}_I] \quad (61)$$

where I denotes quantities at the slave node, and then transferred to the master node (denoted by J) by the cross-product relation

$$\vec{m}_{JG} = \vec{m}_{IG} + \vec{\Delta}_G \times \vec{f}_G \quad (62)$$

The moments for the master node $[m_{JG}]$ are then transformed to the nodal system for that node

$$[\bar{m}_J] = [B]^T[m_{JG}] \quad (63)$$

and entered into the appropriate moment summations in equation (2).

3. ANALYSIS RESULTS

A substantial number of test and demonstration problems have been treated during the development of the analysis and computer program reported here. For the most part the test problems dealt with relatively simple structural configurations having known results and were selected to isolate and test particular aspects of the formulation and program during development. Thirteen such test problems of successively increasing complexity were formally reported in progress reports. In all cases the analyses were pursued until satisfactory comparison with previous results were obtained or at least to the point at which differences between results could clearly be reconciled on the grounds of differences in loadings, boundary conditions, or ambiguities in the results used for comparison purposes. The contents of these test problems are recapitulated briefly in Table 1, and results for two more complicated problems, namely dynamic snap-through for an elastic dome and the response of an elastic-plastic panel under impulsive load are reviewed in the following sections. Results are also presented in subsequent sections which involve the simulation of two actual crash events taken from a previous series of full scale barrier crash tests.

3.1 Test Problems

Results for two test problems are presented in this section. These have been chosen to demonstrate the capabilities of the analysis procedure in terms of fairly difficult problems for which some results are available in the literature. It should be noted however that the results used as references are also based on numerical computations and approximations and are not considered to be "exact" in drawing comparisons.

3.1.1 Dynamic Buckling of a Spherical Shell Segment - This problem concerns the dynamic response of a clamped, axisymmetric, elastic spherical cap under suddenly applied external pressure.

Table 1

SUMMARY OF TEST PROBLEMS

Problem No.	Description	Element Type and Material Properties	Purpose
1	Cantilever beam with suddenly applied lateral tip load	Two-dimensional beam element; elastic properties	Verify small deflection, dynamic solutions
2	Slightly cambered simply supported column with sudden axial end shortening	Two-dimensional beam element; elastic properties	Verify large deflection, dynamic solutions
3	Dynamic loading on circular arch	Two-dimensional beam element; elastic properties	Verify dynamic, snap-through solution
4	Impulsive transverse load on axially restrained beam	Two-dimensional beam elements; elastic-plastic properties	Verify elastic-plastic formulation and interaction with axial constraints
5	Plane stress wave propagation	Membrane triangles; elastic and elastic-plastic properties	Verify membrane formulation for in-plane effects
6	Circular membrane with transverse impulsive load	Membrane triangles; elastic and elastic-plastic properties	Verify membrane formulation for out-of-plane effects
7	Elastic, cantilever plate with suddenly applied lateral tip load	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for small deflections
8	Dynamic loading on circular arch	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for plane snap-through problem
9	Elastic cantilever plate	Membrane-hinge line elements, elastic properties	Verify membrane-hinge line formulation

Table 1 (Concl)

Problem No.	Description	Element Type and Material Properties	Purpose
10	Plane-stress wave propagation	Three-dimensional plate bending elements; elastic properties	Verify membrane damping terms
11	Dynamic loading on circular dome	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for three-dimensional snap-through problem
12	Hood structure subjected to end loadings	Three-dimensional plate; elastic-plastic properties	First "representative" vehicle structure and loading
13	Impulsive load on cylindrical panel	Three-dimensional plate; elastic-plastic properties	Verify plate formulation for three-dimensional elastic-plastic response
14	End-on frame crash simulation	Three-dimensional beam elements; elastic-plastic properties	Simulation of actual stub frame test
15	End-on frame and sheet metal crash simulation	Beams and plates; elastic-plastic properties	Simulation of actual frame and sheet metal test

As indicated in Figure 6, the cap consists of a segment of a sphere of 60 in. radius, with a shallow rise of 0.30 in. and a thickness of 0.08 in. A plan view of the finite element mesh employed in the solution is shown in Figure 7. The mesh consists of 144 plate elements and 82 nodes and covers the complete dome with no constraints or geometric features introduced to enforce an axisymmetric response pattern. This problem has also been treated by Archer and Lange in Ref. 30 by means of a numerical integration of the finite difference equations derived from Marguerre's equations for the axisymmetric response of spherical shells. Displacement time histories for the vertical motion of the center of the cap under an inward pressure of 44 psi are shown in Figure 6 for the present results and for Ref. 30. It is evident that there is fairly close agreement between the two results with the finite element response being slightly delayed (i.e., stiffer) than the results of Ref. 30. It is also apparent that the response lies in the snap-through range since the maximum displacement approaches twice the undeformed rise of the shell. This is also clearly shown in the successive deformed shapes of the shell shown in Figure 8.

3.1.2 Dynamic Response of a Cylindrical Panel - The purpose of this test problem is to compare the results of the present analysis against previously obtained results involving dynamic, elastic-plastic shell buckling response. As indicated in Figure 9, the problem consists of a 120 deg segment of an aluminum cylindrical shell, supported and clamped at all four edges and loaded explosively over a portion of its surface, as shown. Previous experimental and analytical results are available for this problem in the literature, particularly in Refs. 31 and 32.

The mesh geometry employed is illustrated in Figure 10 and consists of 64 elements and 45 nodes representing one-half of the symmetrical shell. The displacement-time history of node 5 is shown in Figure 11 together with corresponding results from Refs. 31 and 32. The explosive load according to Refs. 31 and 32 corresponds to an initial (i.e., impulsive) velocity of 5650 in./sec directed radially inward over the loaded portion of the shell.

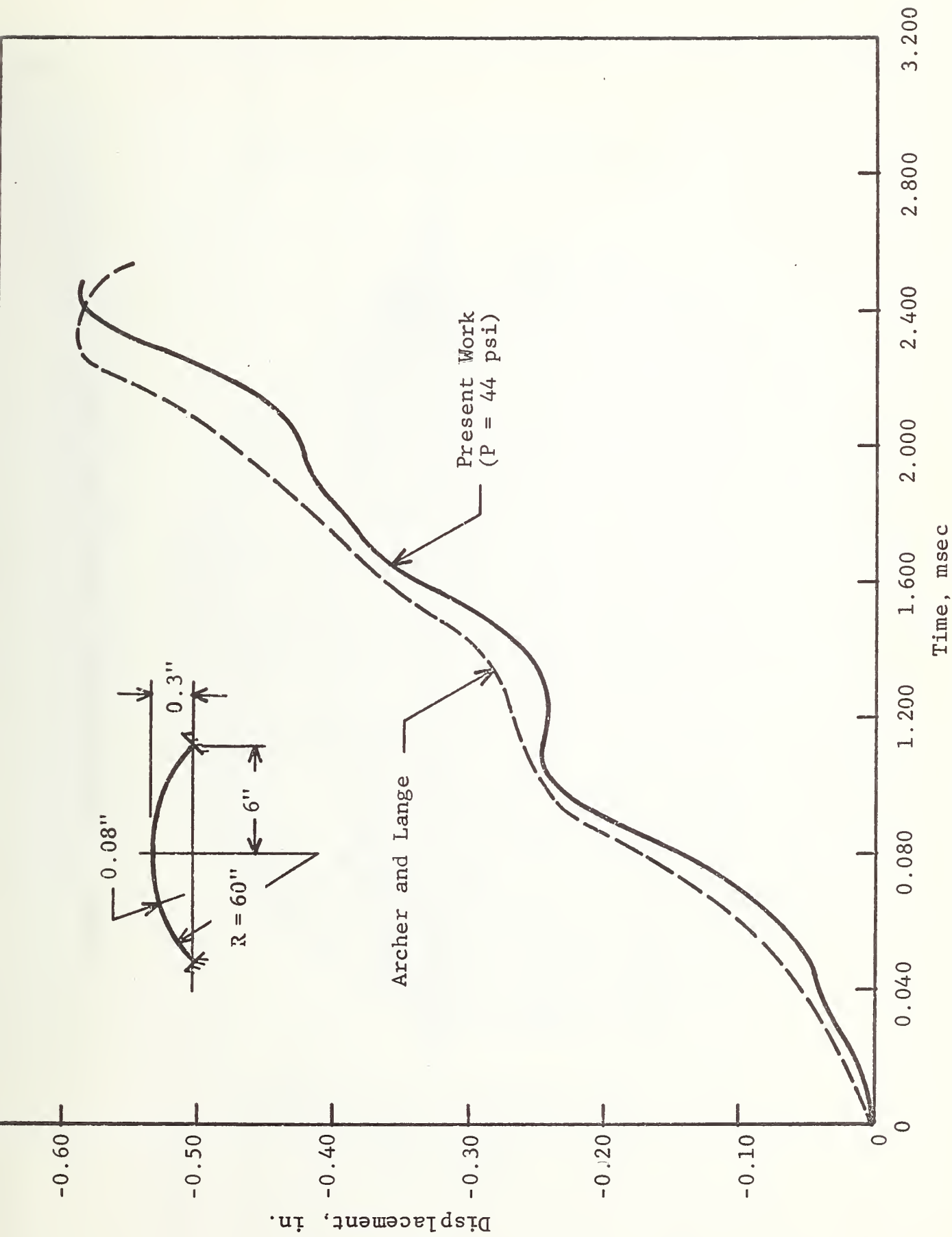


Figure 6 Displacement-Time History of a Clamped Spherical Shell

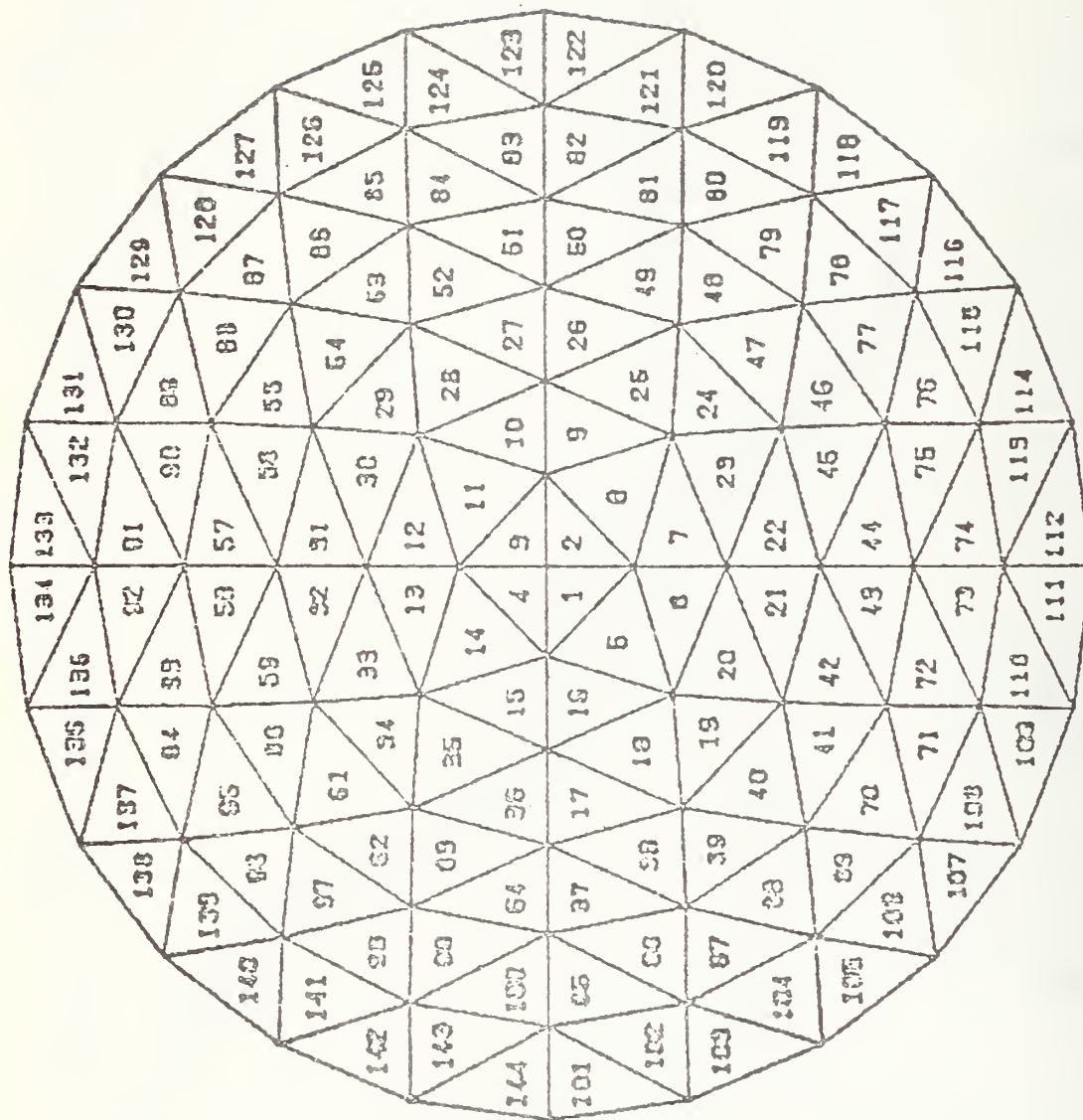


Figure 7 Spherical Shell Finite Element Mesh

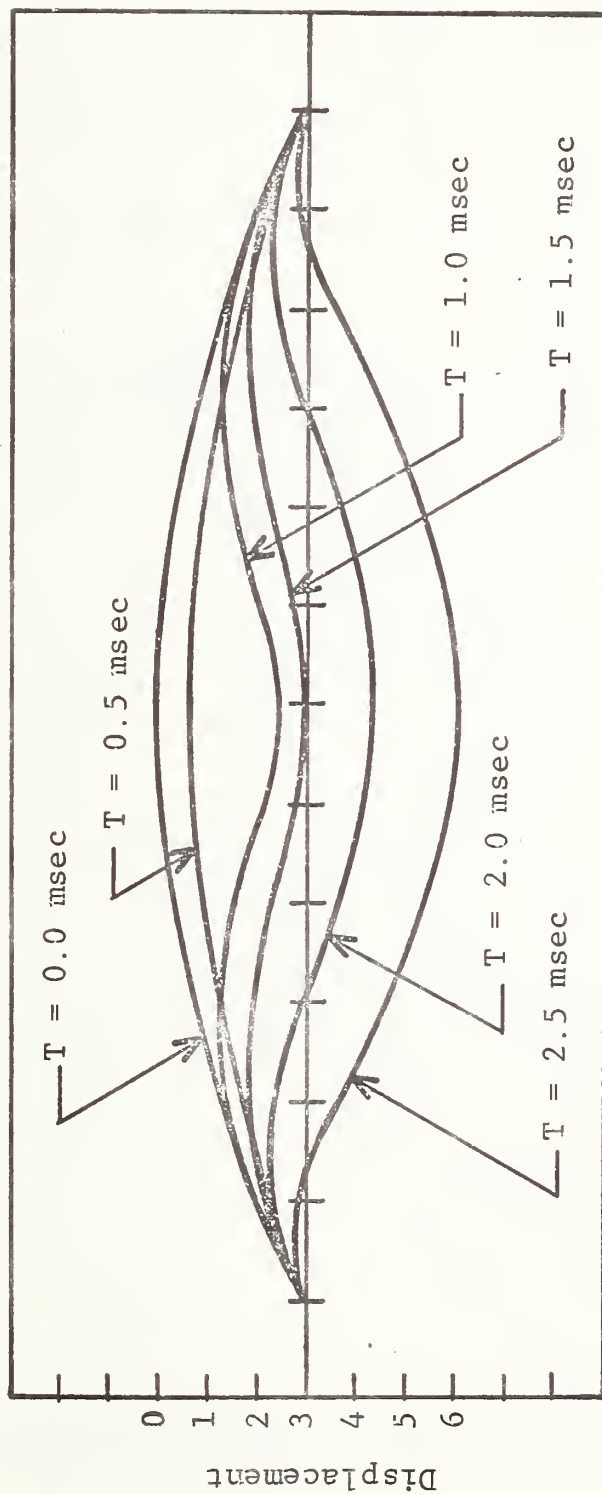
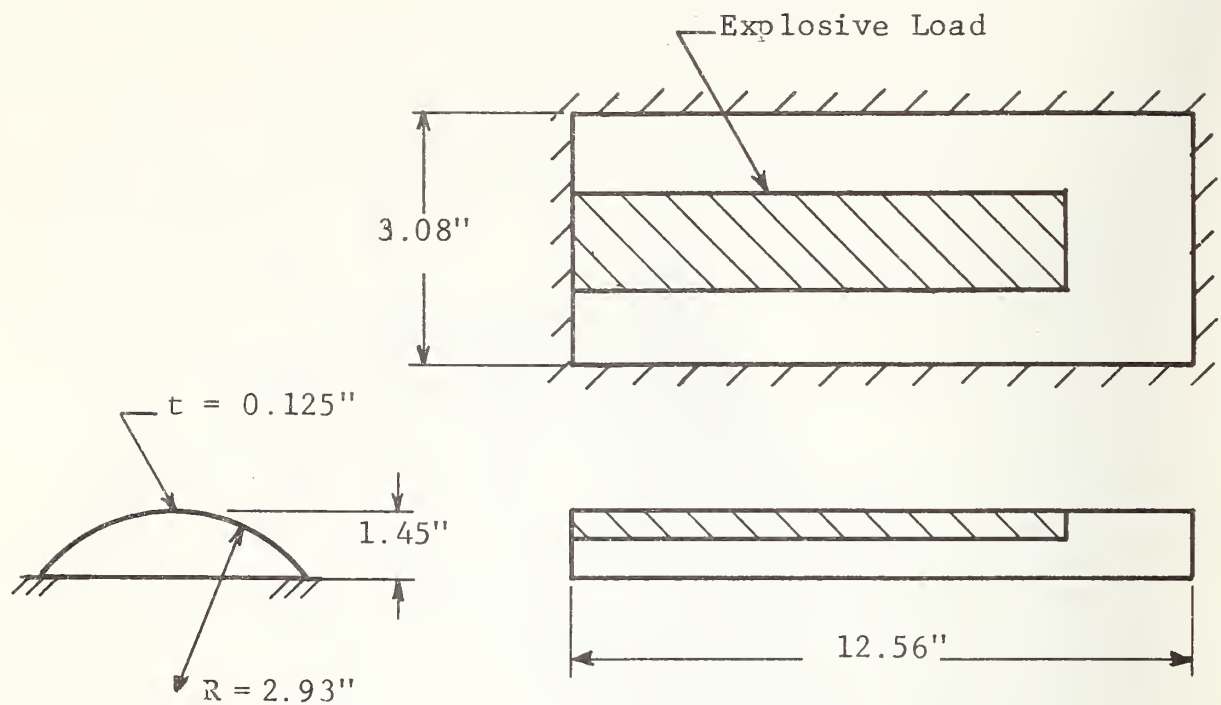
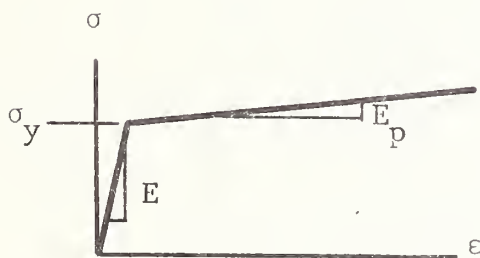


Figure 8 Progressive Deformation Stages of a Clamped Spherical Shell



a. Geometry



b. Properties

AL 6061-T6

$$E = 10.5 \times 10^6 \text{ psi}; \nu = 0.333$$

$$\rho = 2.54 \times 10^{-4} \text{ lb}_f \cdot \text{sec}^2/\text{in}^3$$

$$\sigma_y = 44,000 \text{ psi}; E_p = 0$$

Figure 9 Cylindrical Shell - Geometry and Properties

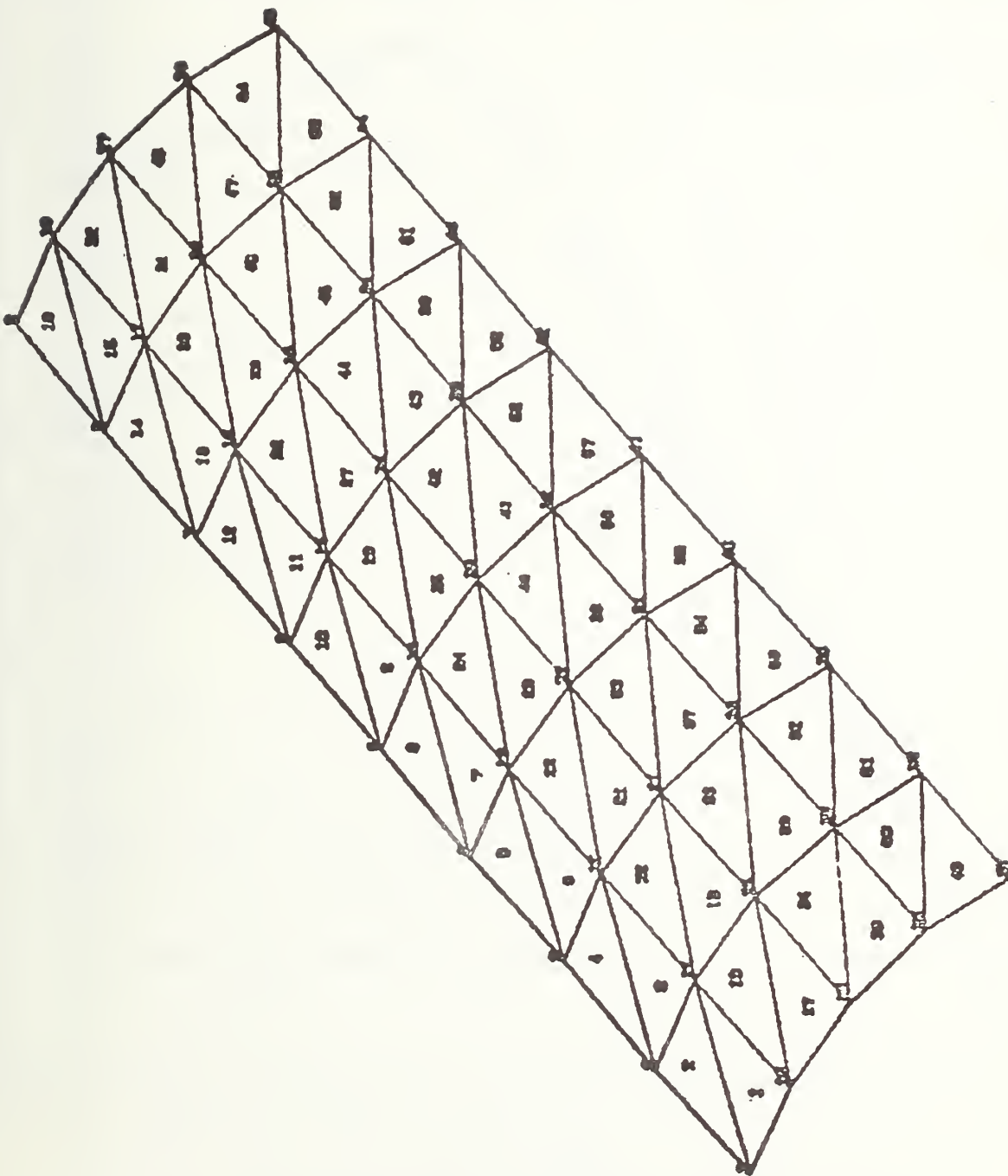


Figure 10 Cylindrical Shell Segment - Undeformed

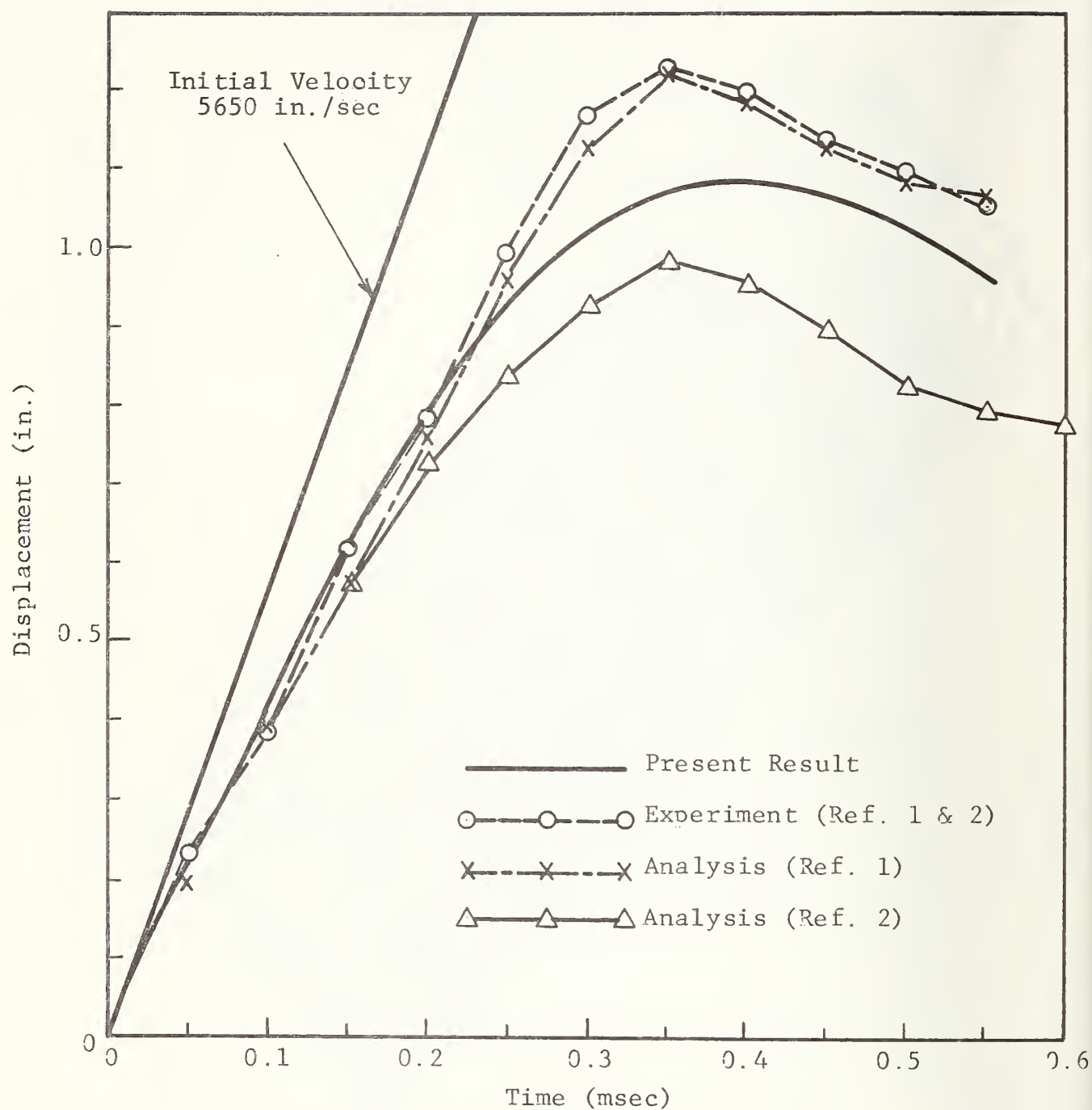


Figure 11 Cylindrical Shell Segment Crown
Displacement versus Time (Node 6)

This corresponds to the constant velocity line in Figure 11 which is tangent to the displacement curve at $t=0$. The present results are bracketed by those of Refs. 31 and 32 with the improved analysis results of Ref. 32 lying below and the experimental results above. The older analysis results from Ref. 31 are very close to the experimental curve but subsequent improvements in that procedure (Ref. 32) result in lower displacements. Improved correlations were obtained in Ref. 32 by introducing yielding boundaries and this was not attempted in the present work.

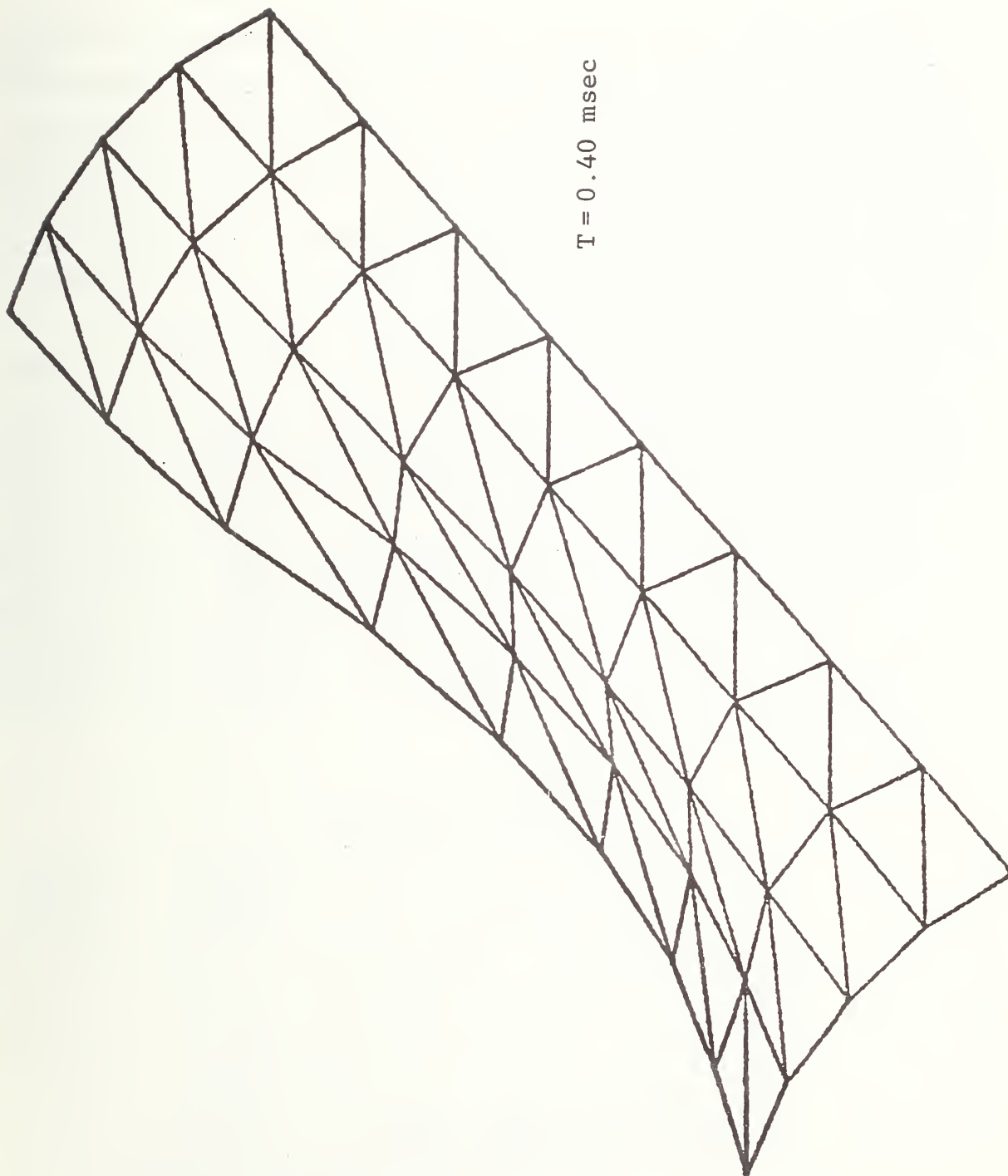
The first part of the velocity history for node 5 is shown in Figure 12. The velocity decreases from its initial value of 5650 in/sec, increases again as the shell rebounds, and then decreases fairly smoothly to zero at approximately 0.4 msec. The corresponding displacement pattern at 0.4 msec is shown in Figure 13. While further efforts could be made to improve the correlations shown in Figure 11, principally in the boundary conditions and in a refined mesh, the present results are considered satisfactory.

3.2 Barrier Test Simulations

The test problems presented throughout the progress reports and in the preceding section are an essential means of assuring that the analysis formulation and computer program are capable of describing the types of phenomena for which they are intended. However, the transition from this class of problems, which are fairly small and idealized, to the analysis of actual vehicle structures in realistic crash events is by no means a trivial step. In the latter case the analysis must deal with much longer duration events and structures of substantially greater size and geometric complexity with realistic, rather than idealized, boundary conditions and connection details. In fact, the proper and efficient application of an analysis procedure to real structures and events is nearly as much a research undertaking, at least initially, as is the original formulation and coding of the analysis.



Figure 12 Cylindrical Shell Segment Crown Velocity versus Time



$T = 0.40 \text{ msec}$

Figure 13 Cylindrical Shell Segment - Deformed

Also, the planning and execution of such analyses has many aspects in common with the planning and execution of full scale crash tests, not the least of which is the acquisition and handling of the large quantities of data which are inherent in either activity.

This section describes the results of our efforts to obtain relatively large scale simulations of actual vehicle crash events. The events which are the objects of these simulations were selected from a series of barrier tests conducted by Dynamic Science (Ref. 33). The series of tests of interest involved frontal impacts at various velocities of a 1968 Plymouth Fury having different portions of the front end structure and machinery removed in different tests as summarized in Table 2. The object of these tests was to isolate and identify the dynamic resistance properties of the various front end components in three groups, namely: frame structure, sheet metal structure and the engine and machinery components. Simulations were developed for two test configurations involving the stub frame alone (test 283-51) and the stub frame and sheet metal together (test 283-53) since these configurations did not involve the added complications associated with the engine components. The test involving sheet metal alone was excluded due to the extreme and unrealistic deformation patterns which appeared in the test results.

Table 2
PLYMOUTH FURY TEST SERIES

Test No.	Configuration	Total Weight (lbs)	Impact Velocity (mph)
283-49	Sheet metal	2665	19.85
283-50	Engine	3914	19.48
283-51	Stub frame	2909	29.64
283-52	Full vehicle	4192	51.45
283-53	Stub frame and sheet metal	2997	43.99

The finite element models developed for these two test configurations represent one-half of the structural components forward of the vehicle fire wall and are based on an assumption of symmetry for both the vehicle and its response modes about the center plane. The frame model consists entirely of beam elements and the frame and sheet metal model contains the frame model, plate elements representing the sheet metal, and additional beam elements representing stiffener sections in the hood. In both cases the appropriate inertial properties of the remainder of the vehicle are provided as added masses at the vehicle center of gravity and are attached to the frontal structure by means of rigid links. Data relating to the tests and the vehicle geometry were obtained from several sources including the following:

- Reports and high-speed motion pictures associated with the test program (obtained through DOT)
- Sheet metal profiles, thicknesses, and material properties obtained from Stanford Research Institute
- Additional measurements and observations of a 1968 Plymouth Fury by IITRI staff.

3.2.1 Stub Frame Impact (Test 283-51) - Plan and side views of the finite element mesh for the stub frame are shown by the dashed (i.e., undeformed) lines in Figures 14 and 15. The model consisted of 22 nodes and 21 beam elements plus a rigid link connecting the rear of the frame at the firewall attachment point to the center of gravity of the vehicle. Additional mass was lumped at the approximate location of the tire connection and also at the vehicle center of gravity in order to introduce the proper inertial characteristics into the analysis. Particular emphasis was placed in accurately modeling the various open and closed wall beam sections that comprise the stub frame. Six different beam cross sections were employed in the model as summarized in Table 3. In all cases the material properties were taken as those of mild steel with an elastic modulus of 30×10^6 psi, a Poisson's ratio of 0.3 and a yield stress of 36,000 psi. Boundary conditions consisted of fixing the fore and aft translatory motion of the nodes in immediate contact with the barrier.

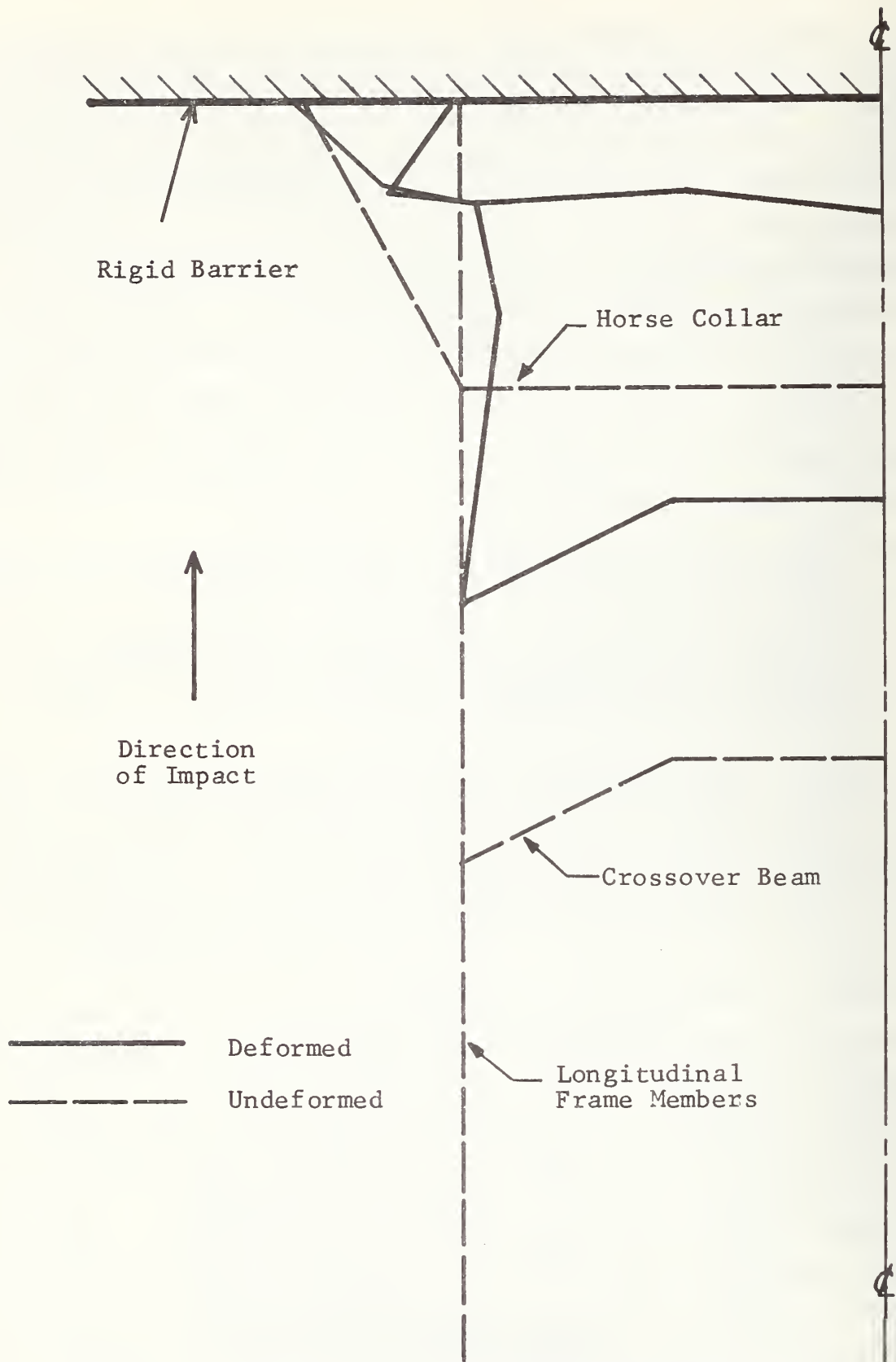


Figure 14 30 mph Stub Frame Impact at 40 msec (Top)

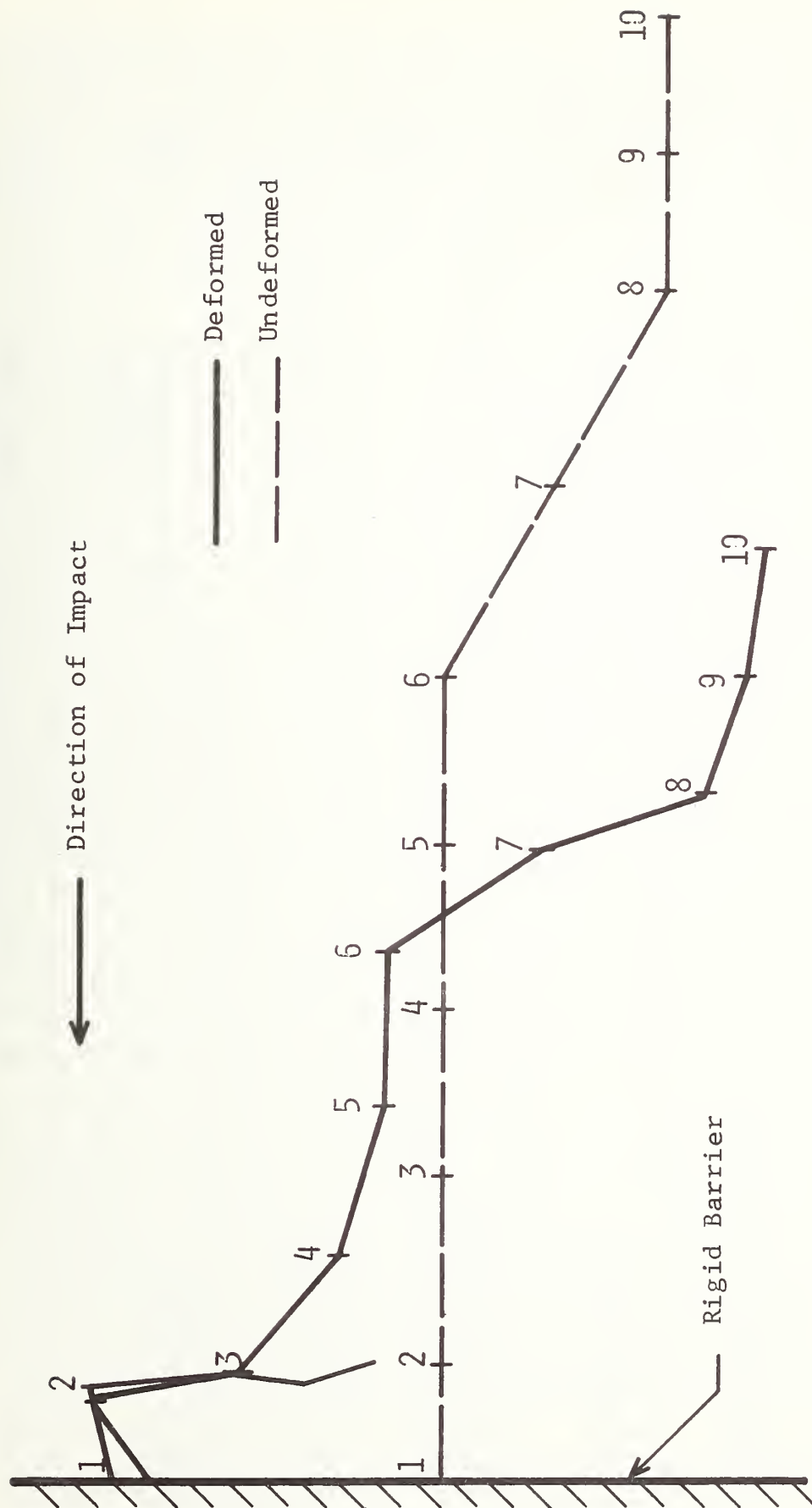








Figure 15 30 mph Stub Frame Impact at 40 msec (Side)

Table 3
STUB FRAME BEAM PROPERTIES

Member Location	Number of Elements	Thickness (in.)	Dimensions (in.)	Shape
Longitudinal Frame (Rear of Crossover Beam)	4	0.085	3 x 3 box	
Crossover Beam	4	0.080	3-1/2 x 3-1/2 box	
Longitudinal Frame (Crossover Beam to Bumper Support)	3	0.085	2 x 6 channel	
Horse Collar Beam	2	0.060	4 x 3 Z-section	
Bumper Support	4	0.060	1-1/2 x 4 channel	
Bumper	4	0.085	2 x 9 bumper	

Rebounding of the frame from the rigid barrier was not taken into account, therefore, once a node came in contact with the barrier, it remained in contact. It should be noted that at the beginning of the solution phase of the analysis it was assumed that all nodes along the bumper were in contact with the barrier. The structure was driven with an initial velocity of 30 mph at all nodes.

Top and side views of the deformed frame at 40 msec after impact are also shown as the solid lines in Figures 14 and 15. It should be noted that the deformed structures are plotted to the same scales as the undeformed mesh and no magnification factors have been introduced. In the plan view, Figure 14, all members show substantial forward motion with bowing in the forward longitudinal frame section and collapse of the bumper supports. Similar behavior is seen in the test films including bowing of the frame members and severe distortions in the bumper supports. In the films the triangle formed by the bumper supports also pivots away from the wall after collapse but this was prevented by the boundary constraints used in the analysis. In the side view, Figure 15, the frame bends throughout the forward portion containing the bumper, bumper supports, and forward frame are seen to have climbed the barrier a distance of approximately 15 in. This same pattern, and particularly the upward motion of the front part of the frame, is clearly evident in the test films although in the films the collapse of the bumper supports is somewhat more severe and the end of the forward frame section comes in contact with the wall. At this point in the motion the frame has begun to rebound from its maximum bent position and the center of mass begins to pivot downward about the point of contact with the wall. At this point the film sequences indicate that this motion is reacted by the rear wheel suspension systems but in the analysis this pivoting continues since the rear wheels were not incorporated in the model, nor were any vertical constraints applied to the center of gravity.

Longitudinal velocity and acceleration at the vehicle center of gravity for 40 msec of motion from the analysis and test results are shown in Figures 16 and 17. As seen in the velocity-time histories, the vehicle has lost only about 20 percent of its initial impact velocity (about 36 percent of its initial kinetic energy). In this test however, this time period contains the maximum vehicle decelerations and barrier loads. A time shift of approximately 10 msec between the analysis and test results is clearly evident in the velocity histories which may be the result of filtering in the test data or differences in zero time between the test and analysis procedures. Whatever the cause, this time shift is found in comparing all corresponding results and will not be commented on further in subsequent discussion. The acceleration time record obtained from the analysis (Figure 17) shows an early deceleration peak of about 10 g in the first 3 msec (note that a signal transit from bumper to center of gravity takes approximately 0.25 msec) and a series of sharp peaks on the order of 30 g between 15 to 27 msec. Peaks in the data are about 22 g at 12 msec and 40 g at 30 msec with a substantial rebound peak at 20 msec. In this case the test data clearly show the effects of filtering. The early peak in the analysis clearly is related to the initial resistance and collapse of the bumper supports while the subsequent peaks are probably attributable to collapse and overall bending of the frame. The analysis also routinely provides contact forces at the barrier but in this case these data were lost beyond the first 5 msec of response.

As indicated in Figure 18 however the early value of the total barrier force was approximately 23,000 lbs (for half of the vehicle) as compared to a value of about 45,000 lbs in the corresponding data (for the full vehicle). This apparent agreement may not be precise however when filtering in the data is taken into account.

3.2.2 Stub Frame and Sheet Metal Impact (Test 283-53) -An isometric view of the finite element mesh for the stub frame and sheet metal model before impact is shown in Figure 19.

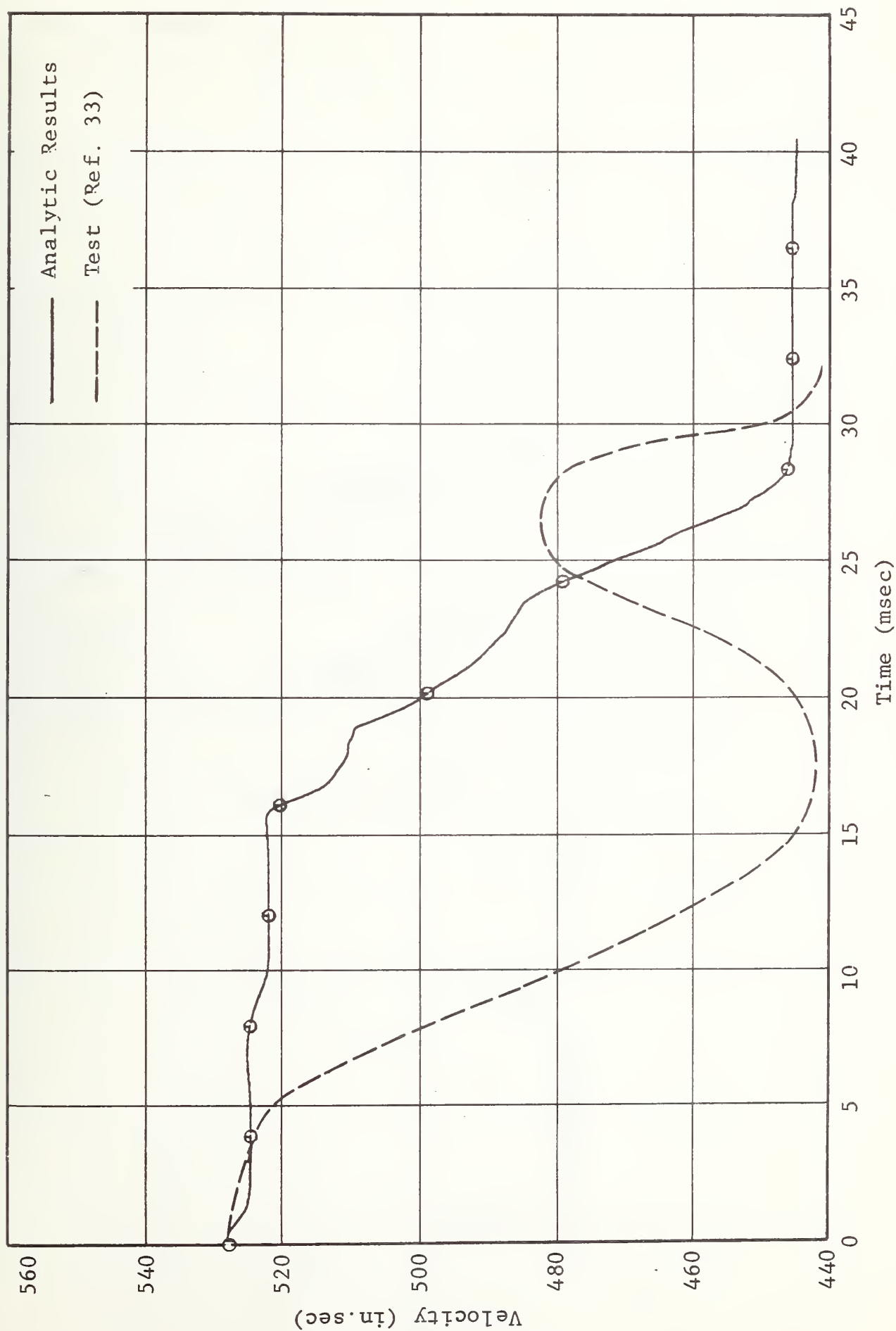


Figure 16 Vehicle Center of Gravity Velocity for 30 mph Stub Frame Test

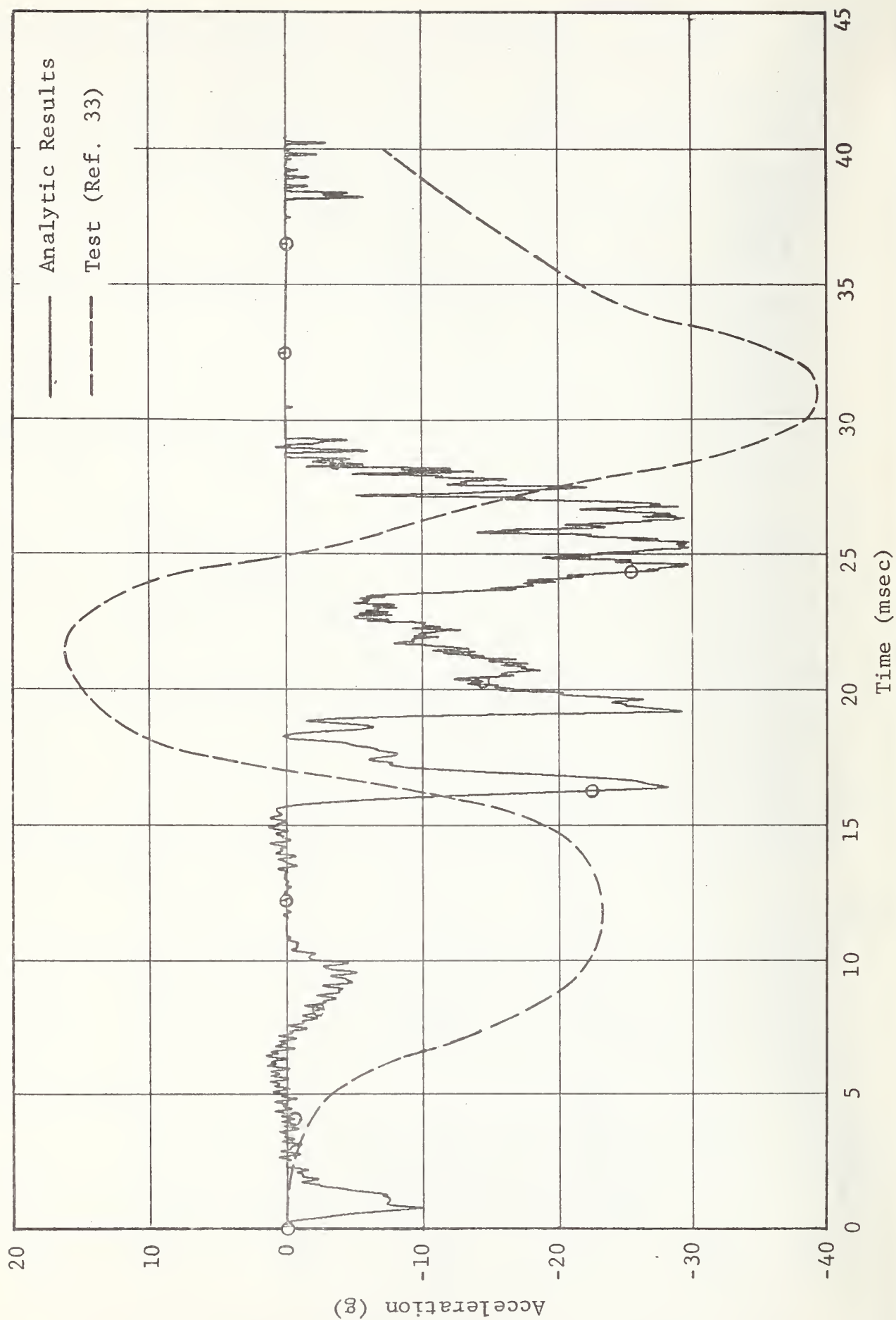


Figure 17 Vehicle Center of Gravity Acceleration for 30 mph Stub Frame Test

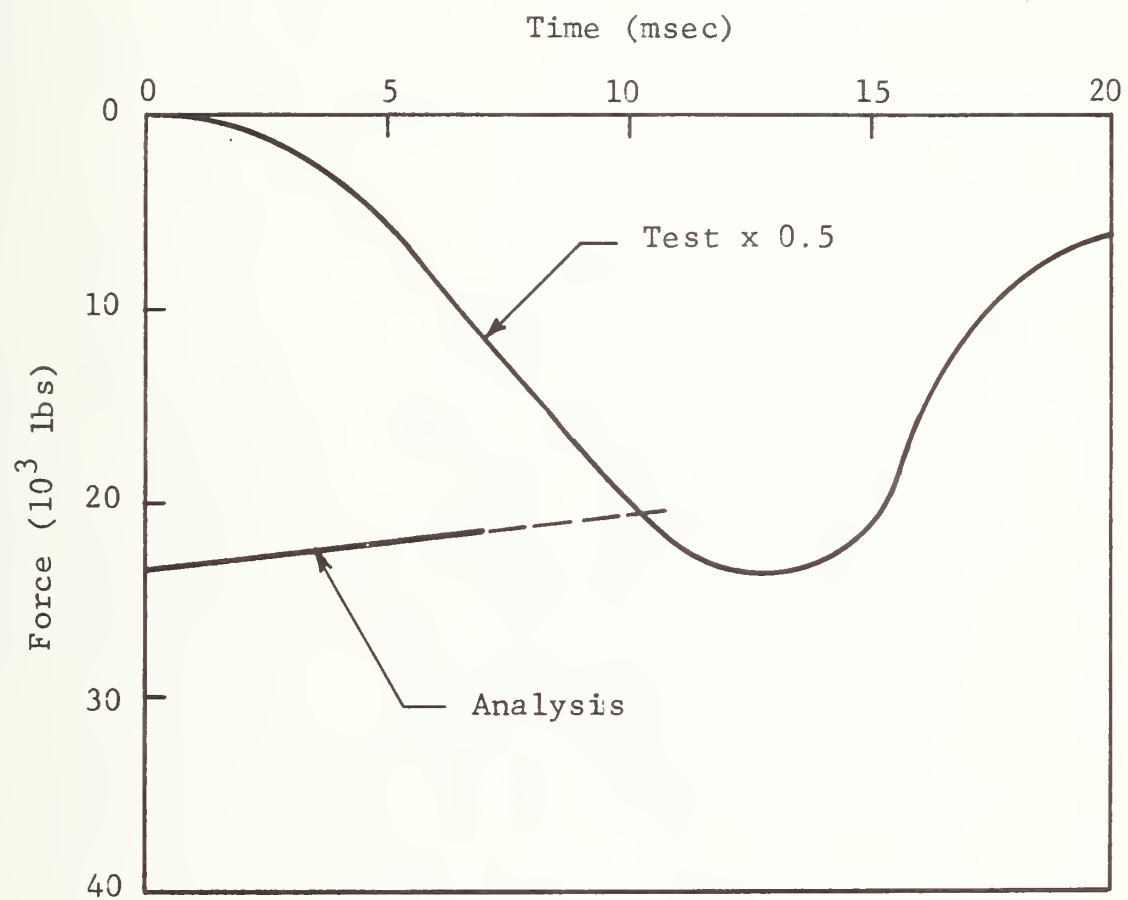


Figure 18 Stub Frame Test - Barrier Force

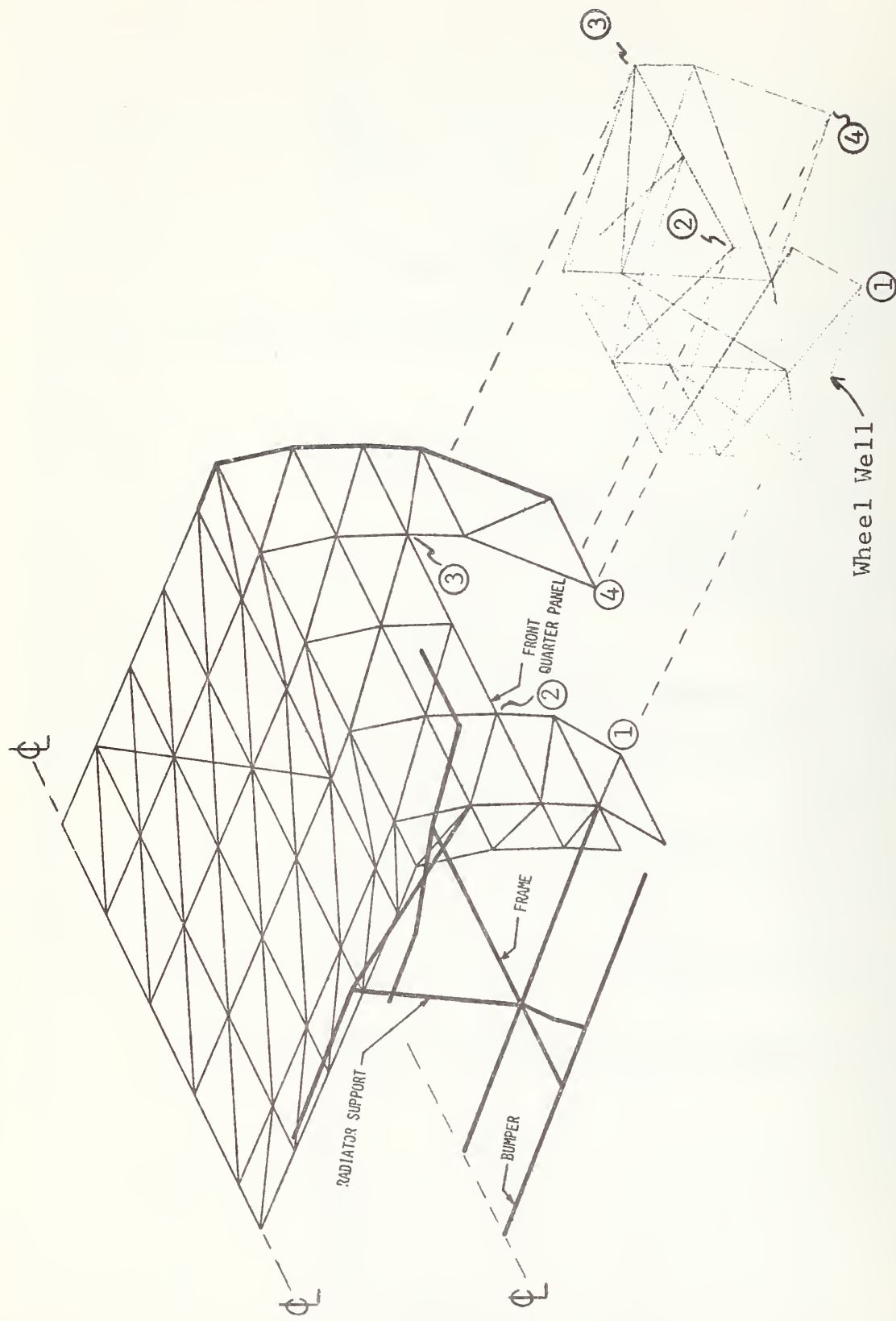


Figure 19 Stub Frame and Sheet Metal Model (Undeformed)

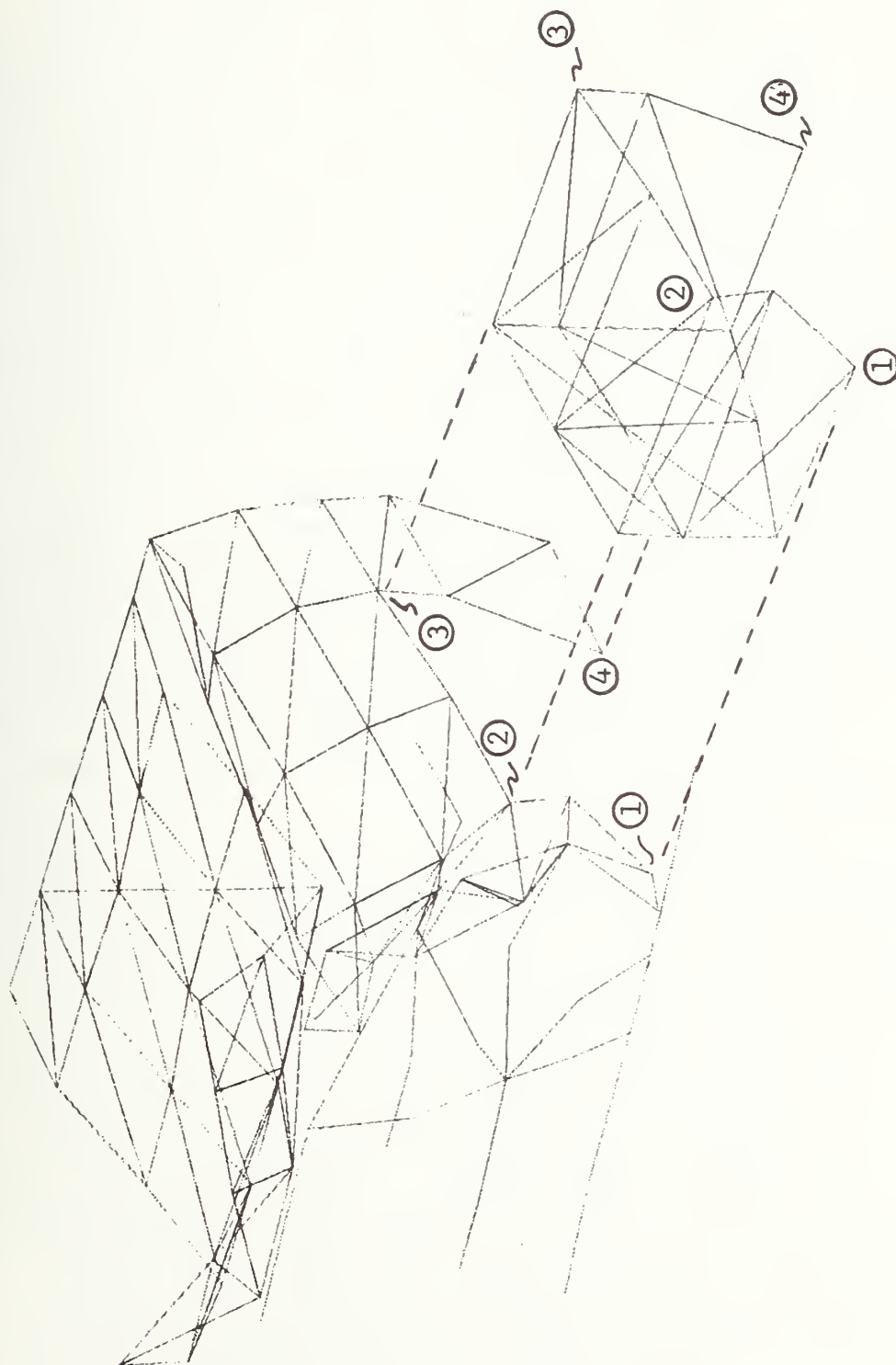


Figure 20 Stub Frame and Sheet Metal Model (Deformed, 20 msec)

This model comprises the left front quarter of the 1968 Plymouth employed in the test. The major components of the model are the hood, left front fender, wheel well and frame and radiator support. The wheel well is shown separately for the sake of clarity in the figure but is actually an integral part of the model and is attached along the lines defined by the match points 1 through 4 indicated in the figure. The thickness of all sheet metal elements was taken as 0.039 in. material. The frame portion of the model is identical to that employed in the stub frame simulation with the addition of the radiator support structure indicated. The hood portion of the model is free to move relative to the fender at their line of intersection, and, in addition to the plate elements shown, contains 2.0 x 1.0 x 0.05 in. hat section stiffeners at reinforcements. As with the stub frame model additional mass was lumped at the approximate location of the wheel suspension and also at the vehicle center of gravity. In this case all points along the rear edge of the fender and hood as well as the end of the frame were attached to the vehicle center of gravity by means of rigid links. The model was driven by an initial longitudinal velocity of 774 in./sec for all nodes with longitudinal constraints applied at nodes in the forward section of the model successively as they came into contact with the barrier.

The deformed shape of the model at 20 msec after impact is shown in Figure 20. At this point the forward motion of the center of gravity is approximately 15 in. and the forward portions of the frame assembly and the first two lines of nodes in the hood and fender are in contact with the barrier. Severe distortions are indicated in the forward portions of the model with the bumper and bumper brackets collapsed downward and the hood and fender collapsing upward and outward respectively. Separation at the hood and fender junction line is clearly evident and the rear portion of the hood has begun to buckle upward away from the fender. The wheel well is not greatly involved although the forward wall has bent to conform to the distortions in the forward section of the fender.

A comparison of longitudinal acceleration time histories at the vehicle center of gravity is given in Figure 21 for 40 msec of motion. The analysis result is similar to that obtained from the analysis of the stub frame configuration in that an early acceleration pulse of approximately 10 g associated with collapse of the bumper support system is apparent. Further the maximum acceleration levels during this phase of the response is approximately 30 g as in the results for the stub frame. In this case however the 30 g response is clearly associated with the impact of fender and hood sheet metal rather than with substantial frame bending. The corresponding test results indicate a peak of about 25 g at a somewhat later time and also clearly show the effects of filtering in comparison to the analysis data.

A similar comparison of total barrier force is provided in Figure 22 in which 50 percent of the test total is compared with the current result obtained from a model of half of the vehicle. The analysis results show an early force level of approximately 40,000 lbs associated with the collapse of the bumper support and frame system which gradually decays for the remainder of the response. The impact of the forward edges of the hood and fender appear as a sudden rise in barrier force at about 6 msec to a level of approximately 80,000 lbs which then decays over the following 10 to 15 msec. This peak is associated with only the first line of nodes in the hood and fender since forces from the subsequent impact of succeeding sheet metal nodes were lost due to a data processing error. In fact, the first three lines of nodes in the fender and hood were involved with the barrier and are included in the results given in Figure 21. In Figure 22, the impact of succeeding nodes would appear as subsequent peaks in the force time history which would appear about the time that the peak due to the previous nodes had decayed substantially. In reality, the progressive contact of the sheet metal with the barrier is a more or less continuous process and the successive impacts of the nodes would amount to a stepwise approximation of this process.

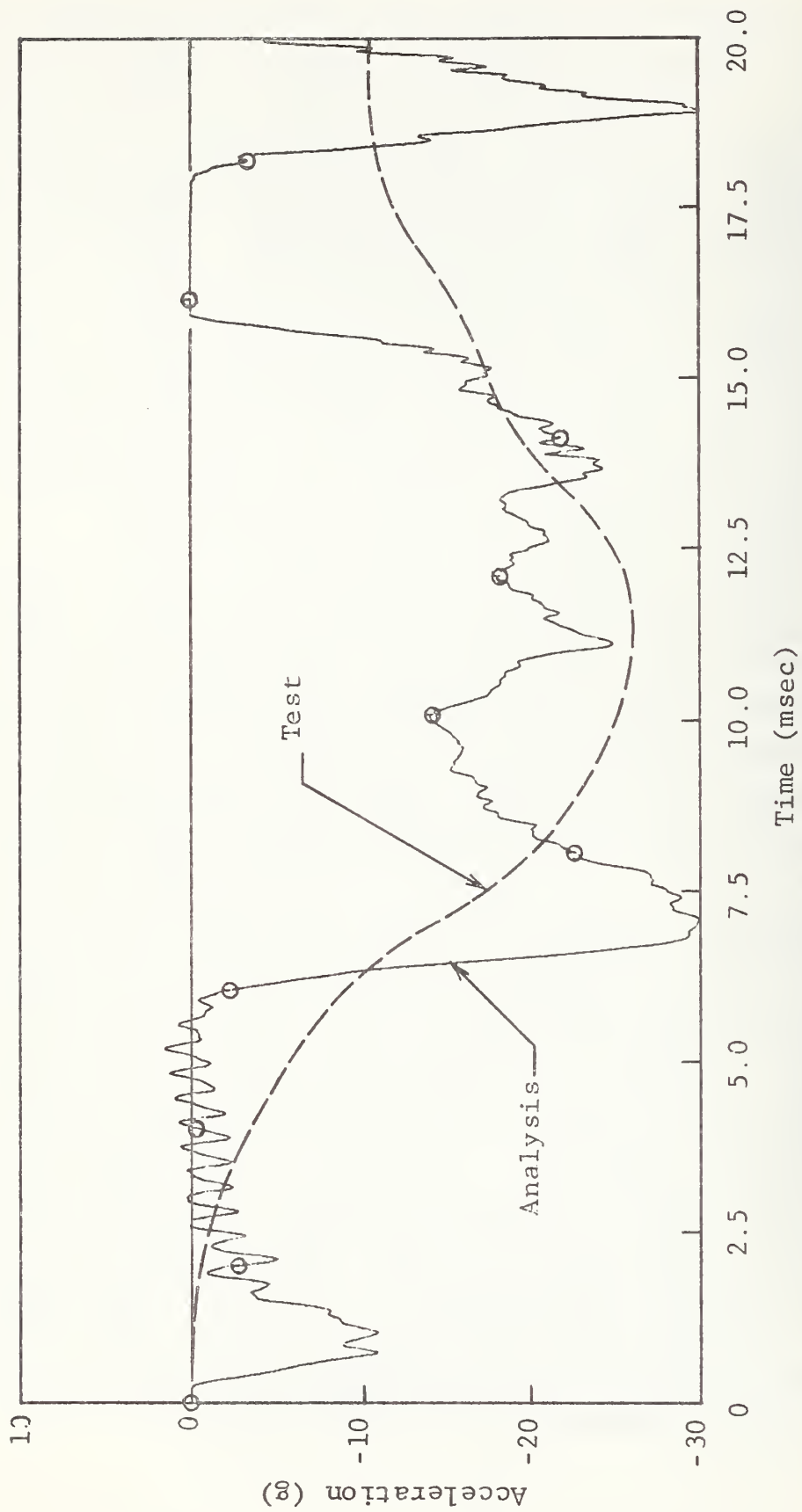


Figure 21 Stub Frame and Sheet Metal Vehicle Center of Gravity Acceleration

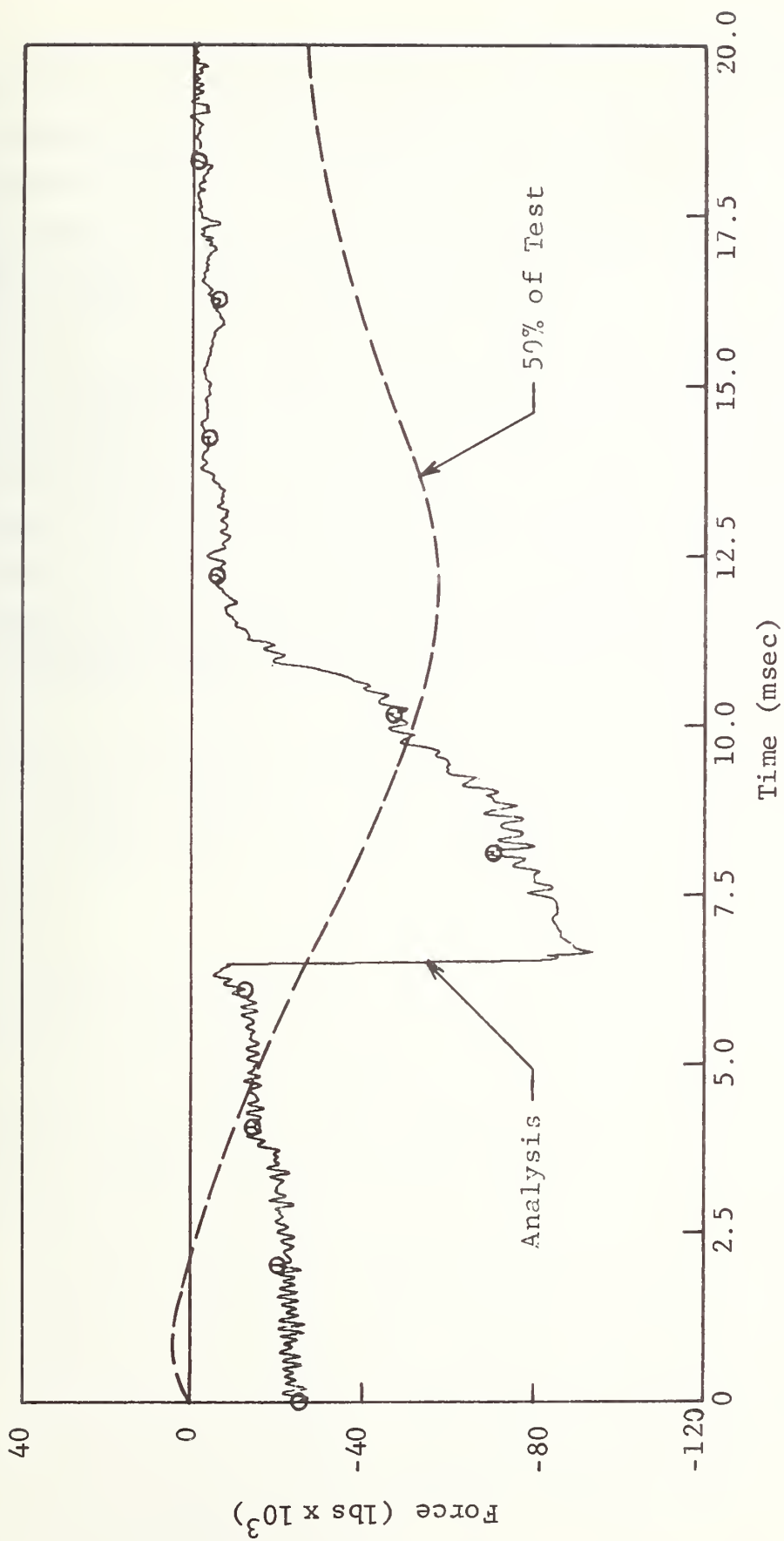


Figure 22 Stub Frame and Sheet Metal Total Barrier Force

A study of the force-time histories for individual nodes at the front of the hood and fender indicates that of the 80,000 lbs total force in the first peak, about 46,000 lbs is derived from collapse of the hood and about 34,000 lbs from the fender.

The test simulations presented here are by no means a complete or exhaustive study even for the limited series of tests which were used for comparison purposes. For a variety of reasons, the results given apply only to the initial phases of the respective crash events and improvements could also be made in the models (the provision of rear wheels or other vertical constraints at the center of gravity, for instance) and in the associated data processing. Despite these limitations, however, the results appear to be a representative portrayal of the phenomena associated with vehicle barrier collisions generally, and a reasonable correlation with the particular test data for which comparisons were attempted.

4. CONCLUSIONS AND RECOMMENDATIONS

This report presents the results of a research project undertaken by IITRI for DOT to develop a finite element computer program for use in the dynamic analysis of vehicle structure, including sheet metal, in a crash environment. This research program involved the following major tasks:

- A technique was developed for the finite element analysis of the dynamic response of plate beam structures involving very large displacements and rotations and elastic-plastic material behavior. The principal feature of this technique involves the decomposition of the element displacement field into rigid body components and deformation components thus allowing the use of a small deflection element formulation in the analysis.
- A computer program was developed incorporating this analysis procedure for beam and plate elements and rigid links together with appropriate time integration procedures and material property descriptions.
- A substantial number of test and demonstration problems were analyzed with this computer program ranging from simple classical solutions for beams and plates through large scale simulations of actual crash tests.

4.1 Conclusions

Based on the work carried out in this research program our principal conclusions are as follows:

(1) The analysis formulation, while somewhat novel, is a formally sound procedure and a proper and useful approximation technique for the dynamic analysis of beam and plate structures involving large deflections and rotations.

(2) The computer program developed is a correct rendering of this analysis technique and provided accurate results with reasonable efficiency in comparison to available known solutions.

(3) Finally, and most importantly, the computer program is readily applicable to realistic vehicle structures and provides credible results for actual crash events as demonstrated by the simulations of the end-on barrier tests.

4.2 Recommendations

In our opinion the analysis and computer program resulting from this research is a substantial improvement over previous attempts at a finite element analysis of vehicle structures and is a potentially valuable addition to the tools available for the study of vehicle crashworthiness. Although many possibilities suggest themselves in terms of further development and refinement in the analysis procedure and computer program (some of which are noted below) the most important and potentially fruitful activity lies in the immediate and continued application of the analysis to actual vehicles and crash events. Such activity will bring the analysis to real vehicle related problems more quickly and will serve as a guide to its further development and application. Thus our primary recommendation is that provision be made for the continued and extensive use of the program in connection with actual vehicle behavior. Some suggestions for such applications as well as for further analysis and program development are as follows:

Applications

- Additional simulation of controlled crash events along the lines of those attempted in this work but extended to include machinery and drive train components and side-on impacts.
- Use of the program in connection with studies of alternate vehicle designs, including potential structural modifications and configuration changes.
- Use of the program to generate structural crush data for subsequent incorporation in the more simple lumped parameter or hybrid class of programs.
- Comparative studies of the deformation and crashworthiness of existing vehicle designs.

Program Development

- Development of suitable approximate moment-curvature relations for sheet metal as a means of improving computational efficiency.

- Incorporation of plate and beam hinges at selected locations in the main version of the program.
- Development of higher order and special hybrid element formulations possibly to include tearing or wrinkling within the element.
- Development of an implicit rather than explicit integration procedure either as an alternate to the present explicit procedure or for selective use for part of the structure or portions of its response history.

APPENDIX A

TRIANGULAR PLATE DISPLACEMENT AND CURVATURE FUNCTIONS

This appendix provides detailed expressions for the transverse displacement and curvature in the triangular plate elements. In terms of triangular coordinates, Zienkiewicz (Ref. 26) gives the displacement of the plate as follows:

$$\begin{aligned}\hat{u}_z = & \beta_1 \zeta_1 + \beta_2 \zeta_2 + \beta_3 \zeta_3 + \beta_4 (\zeta_1 \zeta_2^2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ & + \dots + \beta_9 (\zeta_1^2 \zeta_2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3)\end{aligned}\quad (A1)$$

where $\zeta_1, \zeta_2, \zeta_3$ are the triangular coordinates and $\beta_1, \beta_2, \dots, \beta_9$ are the constants depending on the boundary conditions. We define the nodal rotations as

$$\hat{\theta}_x = \frac{\partial \hat{u}_z}{\partial \hat{y}} ; \quad \hat{\theta}_y = - \frac{\partial \hat{u}_z}{\partial \hat{x}} \quad (A2)$$

Since the displacements are zero at the nodes, the unknowns of equation (A1) can be determined in terms of nodal rotations, and equation (A1) can be expressed as

$$\{\hat{u}_z\} = [N_i] \{\theta\} \quad (A3)$$

where

$$\{\theta\}^T = (\hat{\theta}_{1x} \ \hat{\theta}_{1y} \ \hat{\theta}_{2x} \ \hat{\theta}_{2y} \ \hat{\theta}_{3x} \ \hat{\theta}_{3y})$$

and $N_i, (i=1, \dots, 6)$ are the shape functions given by Meek (Ref. 27). Once the deformation is assumed, the slopes follow

$$\begin{Bmatrix} \frac{\partial \hat{u}_z}{\partial \hat{x}} \\ \frac{\partial \hat{u}_z}{\partial \hat{y}} \end{Bmatrix} = \begin{bmatrix} \frac{\partial N_i}{\partial \hat{x}} \\ \frac{\partial N_i}{\partial \hat{y}} \end{bmatrix} \{\theta\} \quad (A4)$$

and finally the curvatures are given as

$$\begin{Bmatrix} \kappa_{xx} \\ \kappa_{yy} \\ \kappa_{xy} \end{Bmatrix} = \begin{Bmatrix} \frac{\partial^2 \hat{u}_z}{\partial \hat{x}^2} \\ \frac{\partial^2 \hat{u}_z}{\partial \hat{y}^2} \\ 2 \frac{\partial^2 \hat{u}_z}{\partial \hat{x} \partial \hat{y}} \end{Bmatrix} = [\psi] \{\theta\} \quad (A5)$$

where

$$[\psi]^T = \left(\frac{\partial^2 N_i}{\partial \hat{x}^2} \quad \frac{\partial^2 N_i}{\partial \hat{y}^2} \quad 2 \frac{\partial^2 N_i}{\partial \hat{x} \partial \hat{y}} \right)$$

With the element displacement defined by equation (A1) and the slopes by equation (A2), Meek (Ref. 27) gets the following shape functions:

$$\begin{aligned} N_1 &= b_2(\zeta_1^2 \zeta_3 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - b_3(\zeta_1^2 \zeta_2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ N_2 &= a_2(\zeta_1^2 \zeta_3 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - a_3(\zeta_1^2 \zeta_2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ N_3 &= b_3(\zeta_2^2 \zeta_1 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - b_1(\zeta_2^2 \zeta_3 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ N_4 &= a_3(\zeta_2^2 \zeta_1 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - a_1(\zeta_2^2 \zeta_3 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ N_5 &= b_1(\zeta_3^2 \zeta_2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - b_2(\zeta_3^2 \zeta_1 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \\ N_6 &= a_1(\zeta_3^2 \zeta_2 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) - a_2(\zeta_3^2 \zeta_1 + \frac{1}{2} \zeta_1 \zeta_2 \zeta_3) \end{aligned} \quad (A6)$$

When second partial derivatives of these shape functions are obtained, corresponding to the expressions in equation (A5), we get

$$\frac{\partial^2 N_1}{\partial \hat{x}^2} = \frac{1}{(2A)^2} [\zeta_1 b_1 b_3 (b_2 - b_3) + \zeta_2 b_1 b_3 (b_2 - b_3 - 2b_1) + \zeta_3 b_1 b_2 (b_2 - b_3 - 2b_1)] \quad (A7)$$

$$\frac{\partial^2 N_2}{\partial \hat{x}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 [a_2 b_3 (b_2 + 4b_1) - a_3 b_2 (b_3 + 4b_1)] \right. \\ \left. + \zeta_2 b_1 [a_2 b_3 - a_3 (b_3 + 2b_1)] + \zeta_3 b_1 [a_2 (b_2 + 2b_1) - a_3 b_2] \right\}$$

$$\frac{\partial^2 N_3}{\partial \hat{x}^2} = \frac{1}{(2A)^2} [\zeta_1 b_2 b_3 (2b_2 + b_3 - b_1) + \zeta_2 b_1 b_3 (b_3 - b_1) \\ + \zeta_3 b_1 b_2 (b_3 - 2b_2 - b_1)]$$

$$\frac{\partial^2 N_4}{\partial \hat{x}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 [a_3 b_2 (2b_2 + b_3) - a_1 b_2 b_3] + \zeta_2 [a_3 b_1 (4b_2 + b_3) \right. \\ \left. - a_1 b_3 (4b_2 + b_1)] + \zeta_3 [a_3 b_1 b_2 - a_1 b_2 (2b_2 + b_1)] \right\}$$

$$\frac{\partial^2 N_5}{\partial \hat{x}^2} = \frac{1}{(2A)^2} [\zeta_1 b_2 b_3 (b_1 - b_2 - 2b_3) + \zeta_2 b_1 b_3 (b_1 - b_2 + 2b_3) \\ + \zeta_3 b_1 b_2 (b_1 - b_2)]$$

(A7) (Contd)

$$\frac{\partial^2 N_6}{\partial \hat{x}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 [a_1 b_2 b_3 - a_2 b_3 (b_2 + 2b_3)] + \zeta_2 [a_1 b_3 (b_1 + 2b_3) \right. \\ \left. - a_2 b_1 b_3] + \zeta_3 [a_1 b_2 (b_1 + 4b_3) - a_2 b_1 (b_2 + 4b_3)] \right\}$$

$$\frac{\partial^2 N_1}{\partial \hat{y}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 [a_3 b_2 (a_2 + 4a_1) - a_2 b_3 (a_3 + 4a_1)] + \zeta_2 [a_1 a_3 b_2 \right. \\ \left. - a_1 b_3 (a_3 + 2a_1)] + \zeta_3 [a_1 b_2 (a_2 + 2a_1) - a_1 a_2 b_3] \right\}$$

$$\frac{\partial^2 N_2}{\partial \hat{y}^2} = \frac{1}{(2A)^2} [\zeta_1 a_2 a_3 (a_2 - a_3) + \zeta_2 a_1 a_3 (a_2 - a_3 - 2a_1) \\ + \zeta_3 a_1 a_2 (a_2 - a_3 + 2a_1)]$$

$$\frac{\partial^2 N_3}{\partial \hat{y}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 a_2 [b_3 (2a_2 + a_3) - a_3 b_1] + \zeta_2 [a_1 b_3 (a_3 + 4a_2) \right. \\ \left. - a_3 b_1 (a_1 + 4a_2)] + \zeta_3 a_2 [a_1 b_3 - b_1 (a_1 + 2a_2)] \right\}$$

$$\frac{\partial^2 N_4}{\partial \hat{y}^2} = \frac{1}{(2A)^2} [\zeta_1 a_2 a_3 (a_3 + 2a_2 - a_1) + \zeta_2 a_1 a_3 (a_3 - a_1) + \zeta_3 a_1 a_2 (a_3 - 2a_2 - a_1)]$$

$$\frac{\partial^2 N_5}{\partial \hat{y}^2} = \frac{1}{(2A)^2} \left\{ \zeta_1 a_2 [a_2 b_1 - b_2 (a_2 + 2a_2)] + \zeta_2 a_3 [b_1 (a_2 + 2a_3) - a_1 b_2] + \zeta_3 [a_2 b_1 (a_1 + 4a_3) - a_1 b_2 (a_1 + 4a_3)] \right\}$$

$$\frac{\partial^2 N_6}{\partial \hat{y}^2} = \frac{1}{(2A)^2} [\zeta_1 a_2 a_3 (a_1 - a_2 - 2a_3) + \zeta_2 a_1 a_3 (a_1 - a_2 + 2a_3) + \zeta_3 a_1 a_2 (a_1 - a_2)]$$

$$\frac{\partial^2 N_1}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \zeta_1 [(b_2 - b_3)(a_2 b_3 + a_3 b_2) + 4b_1 (a_3 b_2 - a_2 b_3)] + \zeta_2 [(b_2 - b_3)(a_3 b_1 + a_1 b_3) - 4a_1 b_1 b_3] + \zeta_3 [(b_2 - b_3)(a_2 b_1 + a_1 b_2) + 4a_1 b_1 b_2] \right\} \quad (A7) \text{ (Contd)}$$

$$\frac{\partial^2 N_2}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \zeta_1 [(a_2 - a_3)(a_3 b_2 + a_2 b_3) + 4a_1 (a_2 b_3 - a_3 b_2)] + \zeta_2 [(a_2 - a_3)(a_1 b_3 + a_3 b_1) - 4a_1 a_3 b_1] + \zeta_3 [(a_2 - a_3)(a_1 b_2 + a_2 b_1) + 4a_1 a_2 b_1] \right\}$$

$$\frac{\partial^2 N_3}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \zeta_1 [(b_3 - b_1)(a_3 b_2 + a_2 b_3) + 4a_2 b_2 b_3] + \zeta_2 [(b_3 - b_1)(a_3 b_1 + a_1 b_3) + 4b_2 (a_1 b_3 - a_3 b_1)] + \zeta_3 [(b_3 - b_1)(a_2 b_1 + a_1 b_2) - 4a_2 b_1 b_2] \right\}$$

$$\frac{\partial^2 N_4}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \zeta_1 [(a_3 - a_1)(a_3 b_2 + a_2 b_3) + 4a_2 a_3 b_2] + \zeta_2 [(a_3 - a_1)(a_3 b_1 + a_1 b_3) + 4a_2 (a_3 b_1 - a_1 b_3)] + \zeta_3 [(a_3 - a_1)(a_3 b_1 + a_1 b_2) - 4a_1 a_2 b_2] \right\}$$

$$\frac{\partial^2 N_5}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \begin{aligned} &\zeta_1 [(b_1 - b_2)(a_3 b_2 + a_2 b_3) - 4a_3 b_2 b_3] \\ &+ \zeta_2 [(b_1 - b_2)(a_3 b_1 + a_1 b_3) + 4a_3 b_1 b_3] \\ &+ \zeta_3 [(b_1 - b_2)(a_2 b_1 + a_1 b_2) + 4b_3(a_2 b_1 - a_1 b_2)] \end{aligned} \right\}$$

$$\frac{\partial^2 N_6}{\partial \hat{x} \partial \hat{y}} = \frac{1}{2(2A)^2} \left\{ \begin{aligned} &\zeta_1 [(a_1 - a_2)(a_3 b_2 + a_2 b_3) - 4a_2 a_3 b_3] \\ &+ \zeta_2 [(a_1 - a_2)(a_3 b_1 + a_1 b_3) + 4a_1 a_3 b_3] \\ &+ \zeta_3 [(a_1 - a_2)(a_1 b_2 + a_2 b_1) + 4a_3(a_1 b_2 - a_2 b_1)] \end{aligned} \right\} \quad (A7) \text{ (Concl1)}$$

where A is the area of the plate and the constants are related to the coordinates of the node points by:

$$b_1 = \hat{y}_j - \hat{y}_k ; \quad a_1 = \hat{x}_k - \hat{x}_j$$

$$b_2 = \hat{y}_k ; \quad a_2 = -\hat{x}_k$$

$$b_3 = -\hat{y}_j ; \quad a_3 = \hat{x}_j$$

APPENDIX B

COMPUTER PROGRAM INPUT

This appendix presents a description of the computer program input including definition of the input quantities, descriptions of the user supplied driving subroutine, FREEFD, and external data file options and requirements.

1. INPUT FORMAT FOR DOT VEHICLE STRUCTURES ANALYSIS PROGRAM

Card No.	Format	Variable Name	Description
1	20A4	TITLE(20)	80 character run title
2	6I5,E10.6, 2I5	NNODE, NELE, NUMMAT, NUMDIS, MXSTEP, NDGREE, DELT, NCORD, NSECT	NNODE - number of active nodes NELE - number of elements NUMMAT - number of materials NUMDIS - number of displacement specified node points MXSTEP - number of time steps NDGREE - number of degrees of freedom per node (= 6 this version) DELT - time step (sec) NCORD - number of inactive beam pointer nodes NSECT - number of different beam cross section types
3	10I5	KONTRL(10)	Miscellaneous control flags
		KONTRL(1)	= none
		(2)	= none
		(3)	= expansion factor in mesh generation
		(4)	= none
		(5)	= 0;(β parameter)
		(6)	= 0;(iteration parameter)
		(7)	= number of thickness integration points (= 2, if zero)
		(8)	= check point interval
			= 0;no operation
			> 0;made check point
			< 0;restart and make check points
		(9)	= restart checkpoint number
		(10)	= time history data flag
			= 0;no operation
			> 0;make time history file
4.1	I5	MTYPE	MTYPE = material type number

Card No.	Format	Variable Name	Description
4.2	8E10.4	E(1) thru E(8)	Material properties E(1) - density (lb-sec ² /in. ⁴) E(2) - E; modulus of elasticity E(3) - σ_y ; yield stress (psi) 0 ; elastic E(4) - Ep; plastic modulus E(5) - plate thickness (in.) E(6) - σ_{ult} ; ultimate stress (beams) E(7) - ν ; Poissons ratio E(8) - membrane damping as fraction of critical
4.3	8E10.4	E(9) thru E(16)	E(9) - none E(10) - none E(11) - none E(12) - none E(13) - none E(14) - none E(15) - none E(16) - none
Repeat Group 4 NUMMAT Times			
5.1	2I5,2E10.4	NSEG, KLOS, FIYY, FIZZ	NSEG - number of plate segments in beam cross section KLOS - flag for closed wall sections = 0; closed wall = 1; open wall FIYY, - cross section moments of inertia about Y and Z FIZZ principal axes, respectively
5.2	3E10.4	Y(I),Z(I),T(I)	Y(I), - cross section coordinates of point at beginning of segment I (in.) (See figure 5) Z(I) - T(I) - segment thickness (in.)

Card No.	Format	Variable Name	Description
<u>Repeat Card 5.2 NSEG+KLOS Times</u>			
<u>Repeat Group 5 NSECT Times</u>			
<u>Omit Group 5 if NSECT=0</u>			
6	I5, 5X, 3E10.4	N, XC(N), YC(N), ZC(N), PMASS(I,N), I=1,4	Node point data N - node number XC(N) - x YC(N) - y ZC(N) - z } coordinates of node point PMASS - additional (nonstructural) lumped mass at node N (1,N) = translation mass (2,N) = added rotational mass (3,N) = { Ix, Iy, Iz in global (4,N) = components
<u>Repeat Group 6 NNODE + NCORD Times</u>			
Missing nodes are filled in sequentially and spaced uniformly or according to KONTRL(3) parameter.			
7	10I5	M, NODE(9)	Element definition: M - element number Plate element: NODE(1) } plate corner nodes NODE(2) } NODE(3) } NODE(4) } NODE(5) } NODE(6) } NODE(7) } not applicable

Node No.	Format	Variable Name	Description
7			<p>Beam element:</p> <p> NODE(1) } beam end points NODE(2) } NODE(3) } master nodes for NODE(1) NODE(4) } and NODE(2), respectively NODE(5) } - not applicable NODE(6) } - cross section type NODE(7) } - pointer node for initial principal axis \hat{y} NODE(8) } - element type = 0 or 1; plate = 2; beam NODE(9) } - material type = 1 if zero </p>
Repeat Group 7 up to NELE Times			
Missing elements generated sequentially as:			
NODE(I,M+1) = NODE(I,M)+1 NODE(9,M+1) = NODE(9,M)			
Last element in sequence must be read from card			
8	I10, 3E10.4	NODDIS(I), ANGLE(3,I)	<p>Node constraint data:</p> <p>NODDIS - packed word</p> <p>- NABC (right justified) where N = node number and positions A,B,C correspond to X,Y,Z (or $\theta_x, \theta_y, \theta_z$)</p> <p>Displacement constraint:</p> <p>A,B, or C = 0, no constraint = 1, constrained = 2, specified in FREEFD</p>

Card No.	Format	Variable Name	Description
8			Rotation constraint: A,B, or C = 3; no constraint = 4; constrained = 5; specified in FREEFD ANGLE(3) - constraint angles; inoperative this version
		<u>Repeat Group 8 NUMDIS Times</u>	
		<u>Omit Group 8 if NUMDIS = 0</u>	
9	6I10	NPFREQ, NPRU, NPRS, NPRV, NPRP, NPIC	Output control parameters NPFREQ - output frequency (number of iterations) NPRU - number of output nodes NPRS - number of stress output elements NPRV - inoperative NPRP - inoperative NPIC - number of picture output times
9.1	I10	OUT(1)	Displacement output - packed work - NABC (right justified) N = node number A = 1 - x = 2 - y = 3 - z = 4 - θ^x = 5 - θ^y = 6 - θ^z B = 0 - \dot{A}^x = 1 - \dot{A}^y = 2 - \dot{A}^z C = 0
		<u>Repeat Group 9.1 NPRU Times; Omit for NPRU = 0</u>	

Card No.	Format	Variable Name	Description
9.2	I10	IOUT(1)	Stress output Packed work - N; currently inoperative
<u>Repeat Group 9.2 NPRS Times; Omit for NPRS = 0</u>			
9.3	2I10	NPOUT(2)	Picture output NPOUT(1) - time count for picture output NPOUT(2) - picture type =1; displacement =2; velocity =3; acceleration =4; stresses
<u>Repeat Group 9.3 NPIC Times; Omit for NPIC = 0</u>			

2. EXTERNAL FILE REQUIREMENTS

Restart unit - Logical unit 25

Permanent storage for checkpoint data; approximately 27,000 words per checkpoint required. Read and write operations are unformatted FORTRAN type.

Time history unit - Logical unit 22

Permanent storage containing header records and all output quantities (i.e., NPRU, etc.); approximately $\text{NPRU} \times \text{MXSTEP}$ words are required. Read and write operations are unformatted FORTRAN type.

Mesh data file - Logical unit 20

Permanent storage containing initial mesh data (nodes and elements) and displacements for all nodes at selected time steps (NPIC); the approximate number of words required is

$$5 \times \text{NUMEL} + 4 \times (\text{NPIC} + 1) \times \text{NNODE}.$$

Read and write operations are formatted FORTRAN type.

3. FORCING FUNCTION SPECIFICATION

Forcing functions (i.e., displacements on nodal forces as functions of time) are presently provided through a user supplied subroutine FREEFD. The program may be used both for initialization and subsequent driving forces and motions based on the fixed point variable, KONWAV.

KONWAV=2 - initialization
KONWAV=1 - forcing functions

Displacements, velocities, accelerations and external nodal forces are found in UD, UD1, UD2 and FORCD; counted six per node in the sequence $u_x, u_y, u_z, \theta_x, \theta_y, \theta_z$. The program is called at every time step and the current valued time and the time count are contained in variables TIME and NTSTEP. These may be used as arguments in expressions for forces or motions as functions of time.

APPENDIX C

PROGRAM DESCRIPTION

1. GENERAL

This appendix presents a description of the computer program developed in this investigation for the dynamic, large deflection elastic-plastic response of vehicle structures. The program contains beam and triangular plate finite elements and treats the large deflection phenomena by decomposing the element deformations into rigid body and deformation components before computing element forces. The numerical integration technique employed is an explicit Newmark β -method with $\beta = 0$ and a lumped mass approximation for the inertial properties.

As described in this appendix the program consists of approximately 2000 cards and normally executes in 65K words of core storage on the UNIVAC 1108 computer under the EXEC-8, version 27 operating system. A version of the program has been prepared and executed in 240K bytes on an IBM 370/158 system under the H compiler with no optimization. The capacity of the program as described here is approximately 150 nodes, 150 elements, 100 displacement boundary conditions, 5 different sets of material properties and 10 distinct beam cross sections. The solution procedure is completely in core and makes no use of peripheral devices for temporary (i.e., scratch) storage during execution. However, a restart capability is provided via logical unit 25 and output data storage is provided via units 20 and 22. The use of these files is optional and they should be defined as dummy files if not required. The following sections of this appendix provide a brief description of the arrangement and function of the various subroutines and the allocation of the main data storage.

2. SUBROUTINE DESCRIPTIONS

The overall structure and the names and logical relation among subroutines is shown in Figure C-1. The subroutines collect into three main operating groups relating to data input (READIN), preliminary calculations (ASSBLE), and the main solution procedure (SOLVE) plus four small subroutines (utilities) which are used throughout the program. Brief descriptions of the functions of the individual programs follow.

MAIN. The main program is fairly short and contains dimensioning for the main data storage and calls to READIN, ASSBLE, and SOLVE. No computational or control operations are carried out in this program except the calls to these subroutines.

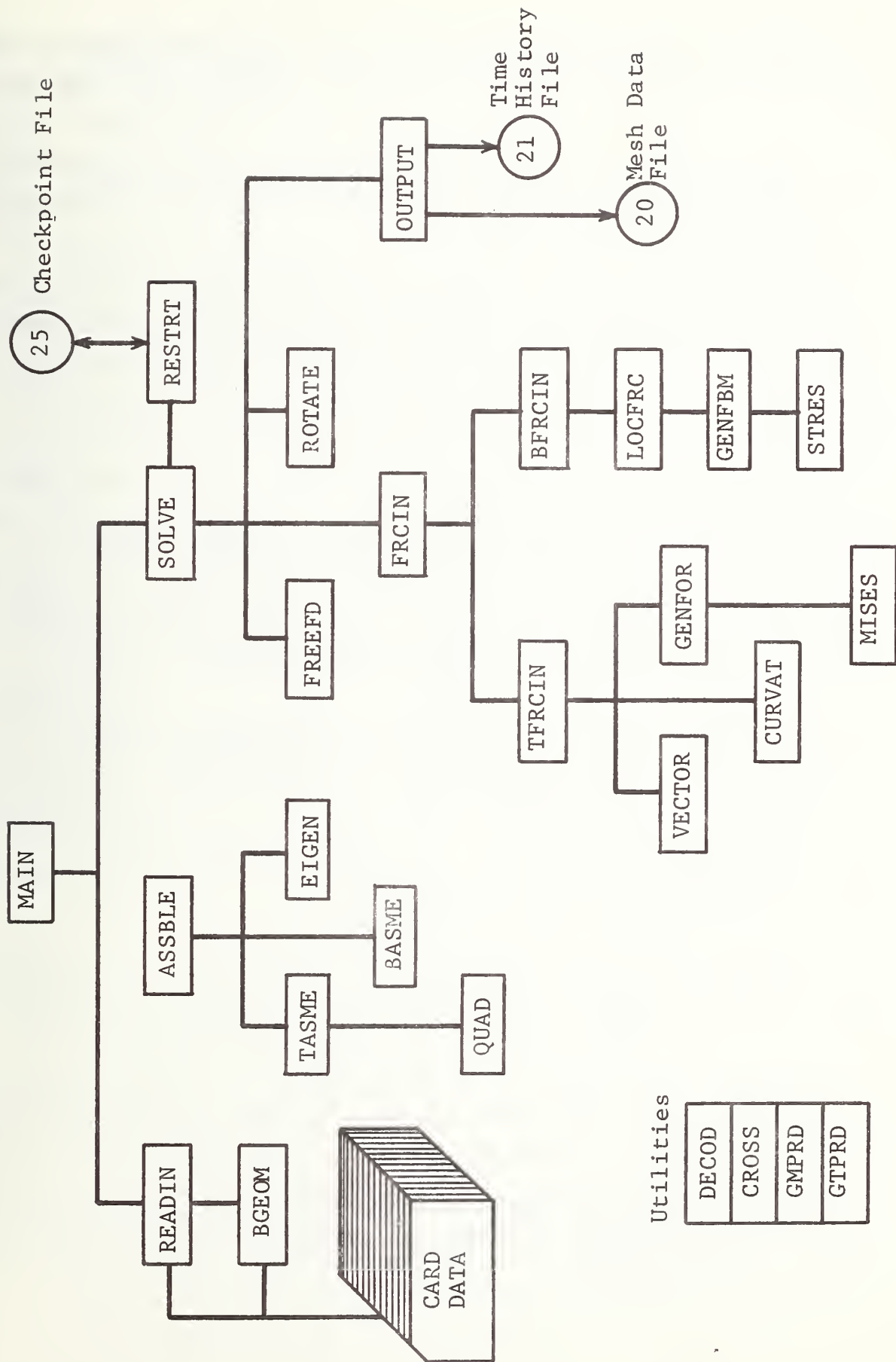
READIN. This program reads all of the input data from cards (except for beam cross section data) according to the input formats described in Appendix B and initializes the data files if required.

BGEOM. Reads data describing the cross section properties of beams and generates certain additional data such as segment lengths and the torsional constants.

ASSBLE. This program initializes the element data storage, assembles the mass matrix and the initial transformations, $[B_0]$ for the nodal coordinate systems and generates the initial values of the pointing vectors $[\bar{n}]$, $[\bar{\eta}]$ and $[\bar{\xi}]$ and the nodal components of the rigid links $[\bar{\Delta}]$.

TASME. Calculates contributions to the lumped mass matrix for plate elements and forms the initial element coordinate system $[E]$ for the plate.

QUAD. An algorithm for lumping the plate rotational inertia at the three defining nodes based on the properties of three quadrilaterals formed in the triangle by extending perpendiculars from the centroid to three sides.



Utilities

DECOD
CROSS
GMPRD
GTPRD

Figure C-1 Computer Program Structure

BASME. Calculates contributions to the lumped mass matrix for beam elements and forms the initial element coordinate system [E].

EIGEN. Generates the principal values and directions for the assembled rotational mass matrix at a node. An IBM utility program found in Ref. 29.

SOLVE. This program contains the numerical integration procedure and the equations of motion (equation (2)). The principal subroutines called by this program are FREEFD for external driving forces or motions, FRCIN for internal nodal forces and OUTPUT which provides printed output and data files.

FREEFD. This program provides nonzero initial conditions or external driving forces or motions at specified nodes and is modified for each application as described in Appendix B.

RESTR. This program generates or reads checkpoint data files which contain sufficient information to restart the solution procedure at specified values of time step.

ROTATE. The program is used to constrain nodal motions in directions other than parallel to the global axes.

FRCIN. Provides internal nodal forces for use in the equations of motion within SOLVE. The program updates the nodal coordinate transformations [B], and calls subroutines TFRCIN for plate forces and BFRCIN for beam forces.

TFRCIN. Calculates plate element deformations in the element coordinate system, calculates nodal forces in the element system and transforms forces to the nodal coordinate system.

VECTOR. Calculates the deformed length of plate element sides and the components of a vector normal to the plate reference surface defined by the displaced positions of the three node points.

CURVAT. Provides algebraic expressions for curvature components at the midpoints of the three element sides as functions of the nodal rotations.

GENFOR. Controls the numerical integration of stresses through the thickness of the plate to obtain internal cross section moments and forces.

MISES. Provides the biaxial, elastic-plastic stress strain algorithm described in Section 2.4.1.

BFRGIN. Calculates beam element deformations in the element coordinate system, calculates nodal forces in the element system, performs the operations associated with the master-slave relations and transforms forces to the nodal coordinate systems.

LOCFRG. Calculates midplane strains and curvatures for the beam and nodal forces and moments in the beam element coordinate system.

GENFBM. Controls the numerical integration of stresses over the cross section of the beam to obtain internal cross section moments and forces.

STRES. Provides the uniaxial, elastic-plastic stress strain algorithm.

OUTPUT. Provides printed output of those quantities selected for output, and updates the time-history and deformed geometry data files.

3. DATA ALLOCATION

The allocation of the main items of data* throughout the program is as follows.

COMMON (blank)

XC(300)	-- Undeformed nodal coordinates
YC(300)	
ZC(300)	
UD(900)	-- Nodal displacement, velocity and acceleration (six per node)
UD1(900)	
UD2(900)	
FORCD(900)	-- External nodal forces
E(16,5)	-- Properties for up to five different materials
IX(9,300)	-- Array of element node correspondences and other element parameters
NODDIS(100)	-- Array of packed word constraint data
NUMNP	-- Number of node points
NUMEL	-- Number of elements
NDGREE	-- Degrees of freedom per node (6)
MUD	-- Not used
NUMDIS	-- Number of displacement points

COMMON/CONTRL/

KONTRL(10)	-- Control flags
------------	------------------

COMMON/DYNAM/

DELT	-- Time increment
TIMEND	-- Maximum time value
MXSTEP	-- Maximum number of time steps
NTSTEP	-- Current step count
TIME	-- Current value of time

*The names of arrays, and number of subscripts differ in some subroutines with correspondence arranged through subroutine calling arguments or explicit EQUIVALENCE statements.

COMMON/XSECT/

NSEG(10) -- Number of segments in beam cross sections
 Y(20,10) -- Beam cross section coordinates
 Z(20,10)
 . XLEN(20,10) -- Beam cross section segment lengths
 T(20,10) and thicknesses
 TORCON(10) -- Torsional constant
 KLOS(10) -- Closed wall section flag
 ABM(10) -- Cross section area and moments of inertia
 FIXX(10) about \hat{x} , \hat{y} and \hat{z} axes
 FIYY(10)
 FIZZ(10)

COMMON/STR/

I1 -- Number of plate cross section integration points
 INDEX(300) -- Array of pointers to first item in element stress array STRS
 STRAIN(45) -- Working storage for stress and strain calculations
 STRESS(45)
 SIDE(900) -- Initial lengths of plate element sides (three per plate element)
 IP(6) -- Plate element side/node counter
 STRS(10000) -- Element stress and strain storage. Format for plates is 3x11x7 with each block of 7 laid out as

$$[\epsilon_{xx}, \epsilon_{yy}, \epsilon_{xy}, \sigma_{xx}, \sigma_{yy}, \sigma_{xy}, \sigma_y]$$
 for 11 points on 3 sides for each element. Format for beams is 2xNPTSx4 with each block of 4 laid out as

$$[\epsilon_{xx}, \sigma_{xx}, \sigma_y, \text{blank}]$$
 for NPTS points at 2 ends of the beam. Note that NPTS=NSEG+KLOS.

COMMON/JUNK/

TL(9,300) -- Components of the [B] transformation laid out by columns of three, three columns per node
 POINT(3,3,300) -- Components of initial plate normal in $[\bar{x}, \bar{y}, \bar{z}]$ system

COMMON/BEAMS/

RMASS(3,3,100) -- Components of assembled rotational
 inertia tensor before diagonalization
 DICOS(3,3,150) -- Components of initial element trans-
 formation [E]
 BAR(1200) -- Beam and rigid link storage laid out as
 $[\bar{\eta}_0; \bar{\xi}_0; \Delta_G^0; \bar{\Delta}] = 12$ words
 at two ends of beam = 24 words per beam
 PMASS(4,100) -- Added nonstructural lumped mass at nodes --
 one translational and three rotational
 components per node
 NSECT -- Number of beam cross section types

COMMON/OUTPA/

NPRU -- Number of displacement output points
 NPRS -- Number of stress output points
 NPFREQ -- Print frequency
 NPIC -- Number of picture output intervals
 OUT(9,40) -- Packed word output list
 NPOUT(2,10) -- Picture output list
 TITLE(20) -- Problem title

APPENDIX D
PROGRAM LISTINGS

```

1.  C
2.  C
3.  C
4.  C
5.  C
6.  C
7.  C
8.  C
9.  C
10. C
11. C
12. C
13. C
14. C
15. C
16. C
17. C
18. C
19. C
20. C
21. C
22. C
23. C
24. C
25. C
26. C
27. C
28. C
29. C
30. C
31. C
32. C
33. C

      * * *   W R E C K E R * * *

      A FINITE ELEMENT COMPUTER PROGRAM FOR THE ANALYSIS OF
      AUTOMOTIVE STRUCTURES UNDER CRASH LOADINGS

      BY   IIT RESEARCH INSTITUTE
      FOR   NHTSA = DOT

      JUNE 1975

      COMMON/WORK/ANGLE(600)
      COMMON XC(300),YC(300),ZC(300),UD(900),UD1(900),UD2(900),
      *FORCD(900),E(16,5),
      *IX(9,300),NODDIS(100),NUMNP,NUMEL,NDGREE,MUD,NUMDIS
      COMMON/CONTRL/KONTRL(10)
      COMMON/DYNAM/DELT,TIMEND,MXSTEP,NTSTEP,TIME,EYMED
      COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6),
      *STRS(10000)
      COMMON /BEAMS/RMASS(3,3,100),DICOS(3,3,150),BAR(1200),
      *PMASS(4,100),NSECT
      DIMENSION XO(900),X1(900),V(900),AO(900),A1(900),SMASS(900)
      DIMENSION FINT(900)
      EQUIVALENCE (X1(1),UD(1)),(V(1),UD1(1)),(AO(1),UD2(1))
      1 CALL READIN(NUMEL,NUMNP,NDGREE,MUD,NUMDIS,NODDIS,
      *E,XC,YC,ZC,IX,ANGLE,UD)
      CALL ASSBLE(NUMEL,NUMNP,NDGREE,IX,XC,YC,ZC,E,SMASS,STRS,STRESS,
      1 STRAIN,INDEX,I1,SIDE,IP)
      CALL SOLVE ( NUMEL,NUMNP,NDGREE,NUMDIS,NODDIS,SMASS,XC,YC,ZC,IX,
      *E,XO,X1,V,AO,A1,FORCD,FINT,ANGLE)
      GO TO 1
      END
00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.

```

SUBROUTINE ROTATE(ANGLE,JNOD,U,ND,KONTRL)
  DIMENSION U(1),ANGLE(1)
  DIMENSION S(3),C(3),UC(3),K(3)
  NDGREE=3
  DO 100 J=1,NDGREE
    S(J)=SIN(ANGLE(J))
    C(J)=COS(ANGLE(J))
    100 K(NDGREE+1-J)=NDGREE#JNOD+1-J
  ENTRY ROTAT(U,KONTRL)
  K1=K(1)
  K2=K(2)
  K3=K(3)
  DO 105 J=1,NDGREE
    IC=K(J)
    105 UC(J)=U(IC)
    GO TO (110,125), KONTRL
  110 GO TO (115,115,120), NDGREE
  115 U(K1)=C(1)*UC(1)+S(1)*UC(2)
    U(K2)=S(1)*UC(1)+C(1)*UC(2)
    RETURN
  120 U(K1)=(C(1)*C(3)-S(1)*S(2)*S(3))*UC(1)+(S(1)*C(3)
    +C(1)*S(2)*S(3))*UC(2)-C(2)*S(3)*UC(3)
    U(K2)=-S(1)*C(2)*UC(1)+C(1)*C(2)*UC(2)+S(2)*UC(3)
    U(K3)=(C(1)*S(3)+S(1)*S(2)*C(3))*UC(1)+(S(1)*S(3)
    +C(1)*S(2)*C(3))*UC(2)+C(2)*C(3)*UC(3)
    RETURN
  125 GO TO (130,130,135), NDGREE
  130 U(K1)=C(1)*UC(1)-S(1)*UC(2)
    U(K2)=S(1)*UC(1)+C(1)*UC(2)
    RETURN
  135 U(K1)=(C(1)*C(3)-S(1)*S(2)*S(3))*UC(1)-S(1)*C(2)*UC(2)
    +C(1)*S(2)*S(3)*UC(3)
    U(K2)=(S(1)*C(3)+C(1)*S(2)*S(3))*UC(1)+C(1)*C(2)*UC(2)
    +S(1)*S(3)-C(1)*S(2)*C(3))*UC(3)
    U(K3)=-C(2)*S(3)*UC(1)+S(2)*UC(2)+C(2)*C(3)*UC(3)
    RETURN
  END

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700

```

1. SUBROUTINE CURVAT(CURV,AREA,XROT,YROT,D,CX,CY)
2. DIMENSION FZ(1),SMX(1),SMY(1),XROT(1),YROT(1),D(1)
3. DIMENSION C(3,6,3),CURV(3,3),EM(3,3)
4. A2=-CX*D(3)
5. A3=CX*D(1)
6. A1=-A2-A3
7. B2=-CY*D(3)
8. B3=-CY*D(1)
9. B1=-B2-B3
10. CONST=(0.5/AREA)**2
11. DO 130 J=1,3
12. GO TO (110,115,120), J
13. 110 CONTINUE
14. C(1,1,1)=0.5*B1*(B3**2-B2**2)
15. C(1,2,1)=0.5*B1**2*(A2-A3)
16. C(1,3,1)=B1*(0.5*B2*(3.*B3-B2)+B3**2)
17. C(1,4,1)=0.5*B1**2*(A1-A3)-B2*(A1*B2+4.*AREA)
18. C(1,5,1)=B1*(0.5*B3*(B3-3.*B2)-B2**2)
19. C(1,6,1)=0.5*B1**2*(A2-A1)+B3*(A1*B3-4.*AREA)
20. C(2,1,1)=0.5*A1**2*(B2-B3)
21. C(2,2,1)=0.5*A1*(A3**2-A2**2)
22. C(2,3,1)=0.5*A1**2*(B1-B3)-A2*(A2*B1-4.*AREA)
23. C(2,4,1)=A1*(0.5*A2*(3.*A3-A2)+A3**2)
24. C(2,5,1)=0.5*A1**2*(B2-B1)+A3*(A3*B1+4.*AREA)
25. C(2,6,1)=A1*(0.5*A3*(A3-3.*A2)-A2**2)
26. C(3,1,1)=0.5*A1*B1*(B2-B3)
27. C(3,2,1)=0.5*A1*B1*(A2-A3)
28. C(3,3,1)=A1*(B1**2-B2**2+0.5*B1*B2)
29. C(3,4,1)=B1*(A1**2-A2**2+0.5*A1*A2)
30. C(3,5,1)=A1*(B3**2-B1**2-0.5*B1*B3)
31. C(3,6,1)=B1*(A3**2-A1**2-0.5*A1*A3)
32. GO TO 125
33. 115 CONTINUE
34. C(1,1,2)=B2*(0.5*B1*(B1-3.*B3)-B3**2)
35. C(1,2,2)=0.5*B2**2*(A3-A2)+B1*(A2*B1-4.*AREA)
36. C(1,3,2)=0.5*B2*(B1**2-B3**2)
37. C(1,4,2)=0.5*B2**2*(A3-A1)
38. C(1,5,2)=B2*(0.5*B3*(3.*B1-B3)+B1**2)
39. C(1,6,2)=0.5*B2**2*(A2-A1)-B3*(A2*B3+4.*AREA)

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900

```



```

40. C(2,1,2)=0.5*A2**2*(B3-B2)+A1*(A1*B2+4.*AREA)
41. C(2,2,2)=A2*(0.5*A1*(A1-3.*A3)-A3**2)
42. C(2,3,2)=0.5*A2**2*(B3-B1)
43. C(2,4,2)=0.5*A2*(A1**2-A3**2)
44. C(2,5,2)=0.5*A2**2*(B2-B1)-A3*(A3*B2+4.*AREA)
45. C(2,6,2)=A2*(0.5*A3*(3.*A1-A3)+A1**2)
46. C(3,1,2)=A2*(B1**2-B2**2+0.5*B1*B2)
47. C(3,2,2)=B2*(A1**2-A2**2+0.5*A1*A2)
48. C(3,3,2)=0.5*A2*B2*(B3-B1)
49. C(3,4,2)=0.5*A2*B2*(A3-A1)
50. C(3,5,2)=A2*(B2**2-B3**2+0.5*B2*B3)
51. C(3,6,2)=B2*(A2**2-A3**2+0.5*A2*A3)
52. GO TO 125
53. 120 CONTINUE
54. C(1,1,3)=B3*(0.5*B1*(3.*B2-B1)+B2**2)
55. C(1,2,3)=0.5*B3**2*(A3-A2)-B1*(A3*B1+4.*AREA)
56. C(1,3,3)=B3*(0.5*B2*(B2-3.*B1)-B1**2)
57. C(1,4,3)=0.5*B3**2*(A1-A3)+B2*(A3*B2+4.*AREA)
58. C(1,5,3)=0.5*B3*(B2**2-B1**2)
59. C(1,6,3)=0.5*B3**2*(A1-A2)
60. C(2,1,3)=0.5*A3**2*(B3-B2)-A1*(A1*B3+4.*AREA)
61. C(2,2,3)=A3*(0.5*A1*(3.*A2-A1)+A2**2)
62. C(2,3,3)=0.5*A3**2*(B1-B3)+A2*(A2*B3+4.*AREA)
63. C(2,4,3)=A3*(0.5*A2*(A2-3.*A1)-A1**2)
64. C(2,5,3)=0.5*A3**2*(B1-B2)
65. C(2,6,3)=0.5*A3*(A2**2-A1**2)
66. C(3,1,3)=A3*(B3**2-B1**2+0.5*B1*B3)
67. C(3,2,3)=B3*(A3**2-A1**2+0.5*A1*A3)
68. C(3,3,3)=A3*(B2**2-B3**2+0.5*B2*B3)
69. C(3,4,3)=B3*(A2**2-A3**2+0.5*A2*A3)
70. C(3,5,3)=0.5*A3*B3*(B1-B2)
71. C(3,6,3)=0.5*A3*B3*(A1-A2)
72. 125 CONTINUE
73. DO 130 I=1,3
74. 130 CURV(I,J)=CONST*(C(1,1,J)*XROT(1)+C(1,2,J)*YROT(1)+C(1,3,J)
75. 1*XROT(2)+C(1,4,J)*YROT(2)+C(1,5,J)*XROT(3)+C(1,6,J)*YROT(3))
76. RETURN
77. ENTRY NODALF(EM,SMX,SMY,FZ,AREA)
78. CONST=CONST*AREA/3.
00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800

```

79.
80.
81.
82.
83.
84.
85.
86.
87.
88.
89.
90.
91.
92.
93.

```

DO 140 J=1,3
  SMX(J)=0.
  SMY(J)=0.
DO 140 K=1,3
DO 140 I=1,3
  SMX(J)=SMX(J)+CONST*C(I,2*J-1,K)*EM(I,K)
140 SMY(J)=SMY(J)+CONST*C(I,2*J,K)*EM(I,K)
  SMXT=-SMX(1)-SMX(2)-SMX(3)
  SMYT=SMY(1)+SMY(2)+SMY(3)
  CONST=0.5/AREA
  FZ(2)=CONST*(A2*SMXT+B2*SMYT)
  FZ(3)=CONST*(A3*SMXT+B3*SMYT)
  FZ(1)=-FZ(2)-FZ(3)
  RETURN
END

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.

```
SUBROUTINE DECOD(ML,JJ)
DIMENSION JJ(1)
MK=ML
JJ(4)=MK/1000
MK=MK-JJ(4)*1000
DO 10 I=1,2
J=10*(3-I)
JJ(I)=MK/J
10 MK=MK-JJ(I)*J
JJ(3)=MK
RETURN
END
```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200

1.
2.
3.
4.
5.

SUBROUTINE PRESS(IP,P)
COMMON/DYNAM/DELT,TIMEND,MXSTEP,NTSTEP,TIME,EYMED
P=100.
RETURN
END

00000100
00000200
00000300
00000400
00000500

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.

SUBROUTINE QUAD(X1,Y1,X2,Y2,X3,Y3,A,XI,YI)
C CALCULATES THE MOMENT OF INERTIA OF THE SUBTRIANGLES OF AN ELEMENT
C
C
    ICOUNT=0
    E1=0.5
    E2=0.5
    E3=0.0
    40 XI=A*((Y1*E1+Y2*E2+Y3*E3)**2)/3.+XI
    YI=A*((X1*E1+X2*E2+X3*E3)**2)/3.+YI
    ICOUNT=ICOUNT+1
    GO TO (10,20,30),ICOUNT
    10 E1=0.0
    E3=0.5
    GO TO 40
    20 E1=0.5
    E2=0.0
    GO TO 40
    30 RETURN
    END
00000100
00000200
00000000
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.

```

SUBROUTINE GMPRD(A,B,R,N,M,L)
DIMENSION A(1),B(1),R(1)
IR=0
IK=-M
DO 10 K=1,L
IK=IK+M
DO 10 J=1,N
IR=IR+1
JI=J-N
IB=IK
R(IR)=0.
DO 10 I=1,M
JI=JI+N
IB=IB+1
10 R(IR)=R(IR)+A(JI)*B(IB)
RETURN
END

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700

```

1. SUBROUTINE GENFOR (EXT,CURV,AM,FX,E,H,I1,8TRS
2. )
3. C
4. C
5. C
6. C
7. C
8. C
9. C
10. C
11. C
12. C
13. C
14. C
15. C
16. C
17. C
18. C
19. C
20. C
21. C
22. C
23. C
24. C
25. C
26. C
27. C
28. C
29. C
30. C
31. C
32. C
33. C
34. C
35. C
36. C
37. C
38. C
39. C

PROGRAM FOR THRUST AND MOMENT FROM EXTENSION AND CURVATURE BY
NUMERICAL INTEGRATION AT I1 POINTS
DIMENSION E(1),EPS(3,5),S(3,5),STRS(1),Y(5),AM(3),FX(3),
SEXT(3),CURV(3)
I1=GE.2.AND.I1.LE.5 NOS. OF POINTS IN SECTION
FI1=I1
DH = H/(FI1-1.)

CALCULATE CROSS-SECTION STRAIN AT 3 DIFFERENT DIRECTIONS ID OF
EACH ELEMENT

DO 1 I = 1,I1
FI = I-1
Y(I) = .5*H=FI*DH

ID = DIRECTION

DO 2 ID = 1,3
EPS(ID,I) = EXT(ID)-Y(I)*CURV(ID)
2 CONTINUE
IND = 7*(I-1)+1
CALL MISES (E,S(1,I),STRS(IND),EPS(1,I))
1 CONTINUE

FORM FORCES AND MOMENTS FOR EACH DIRECTION ID

DO 3 ID = 1,3
FX(ID) = 0.
AM(ID) = 0.
DO 3 I = 2,I1
FX(ID) = FX(ID)+S(ID,I)+S(ID, I-1 )
AM (ID) = AM(ID)+S(ID,I)*(Y(I-1)+2.*Y(I))+S(ID, I-1 )*(2.*Y(I-1)
S+Y(I))
3 CONTINUE
DO 4 ID = 1,3
FX (ID) = .5*DH*FX(ID)
AM (ID) = AM(ID)*DH/6.

```

40.
41.
42.

4 CONTINUE
RETURN
END

00004000
00004100
00004200


```

1. SUBROUTINE MISES(E,S,STRS,EPS)
2.   DIMENSION E(1),S(1),EPS(1),STRS(1)
3.   DIMENSION D(3),SO(3)
4.   DO 100 J=1,3
5.     D(J)=EPS(J)-STRS( J)
6.     SO(J)=STRS( J+3)
7.   C ELASTIC STRESSES
8.     STX=SO(1)+E(2)/(1.-E(7)**2)*(D(1)+E(7))*D(2))
9.     STY=SO(2)+E(2)/(1.-E(7)**2)*(D(2)+E(7))*D(1))
10.    STXY=SO(3)+E(2)/(1.-E(7)**2)*D(3)/2.
11.    IF(E(3).EQ.0.) GO TO 125
12.  C EQUIVALENT STRESSES
13.    SIGNED=SQRT(STX**2-STX*STY+STY**2+3.*STXY**2)
14.    IF(SIGNED.LE.STRS( 7)) GO TO 125
15.    CONST=2.*E(4)*(1.-E(7))/(3.*(E(2)-E(4)))
16.    EQSTS=(STRS(7)+CONST*SIGNED)/(1.+CONST)
17.    DLB=(SIGNED/EQSTS-1.)/3.
18.    CON=1./(1.+3.*DLB)
19.    STRS(7)=EQSTS
20.    S(1)=(STX+DLB*(STX+STY))*CON
21.    S(2)=(STY+DLB*(STX+STY))*CON
22.    S(3)=STXY*CON
23.    GO TO 20
24. 125 S(1)=STX
25.    S(2)=STY
26.    S(3)=STXY
27. 20 DO 10 I=1,3
28.    STRS(I)=EPS(I)
29. 10 STRS(I+3)=S(I)
30.    RETURN
31.  END

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.

```

SUBROUTINE GTPRD(A,B,R,N,M,L)
DIMENSION A(1),B(1),R(1)
IR=0
IK=N
DO 10 K=1,L
IJ=0
IK=IK+N
DO 10 J=1,M
IB=IK
IR=IR+1
R(IR)=0
DO 10 I=1,N
IJ=IJ+1
IB=IB+1
10 R(IR)=R(IR)+A(IJ)*B(IB)
RETURN
END

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700

```

1. SUBROUTINE STRES(STRAIN,SIGNEW,E,STRS)
2.
3. C
4. C
5. C
6. C
7. C
8. C
9. C
10. DIMENSION E(1),STRS(1)
11. EPSOLD=STRS(1)
12. SIGOLD=STRS(2)
13. SIGYLD=STRS(3)
14. DEPS=STRAIN-EPSOLD
15. SIGNEW=SIGOLD+E(2)*DEPS
16. IF(E(3).EQ.0.) GO TO 1
17. ASIGN=ABS(SIGNEW)
18. ASIGY=ABS(SIGYLD)
19. IF(ASIGN.LE.ASIGY) GO TO 1
20. DSIGE=ASIGY-ABS(SIGOLD)
21. IF($IGNEW*SIGOLD.LT.0.) DSIGE=ASIGY+ABS(SIGOLD)
22. DEPSE=SIGN(DSIGE/E(2),DEPS)
23. SIGNEW=SIGN(ASIGY,SIGNEW)+E(4)*(DEPS-DEPSE)
24. IF(E(6).NE.0..AND.ABS(SIGNEW).GT.E(6))
25. * SIGNEW=SIGN(E(6),SIGNEW)
26. STRS(3)=SIGNEW
27. 1 STRS(1)=STRAIN
28. STRS(2)=SIGNEW
29. RETURN
30. END

```

```

0000100
0000200
0000300
0000400
0000500
0000600
0000700
0000800
0000900
0001000
0001100
0001200
0001300
0001400
0001500
0001600
0001700
0001800
0001900
0002000
0002100
0002200
0002300
0002400
0002500
0002600
0002700
0002800
0002900
0003000

```

```

1. SUBROUTINE EIGEN(A,R,N,MV)
2. DIMENSION A(1),R(1)
3. RANGE=1.0E-6
4. IF(MV=1)10,25,10
5. 10 IQ=N
6. DO 20 J=1,N
7.  IQ=IQ+N
8. DO 20 I=1,N
9.  IJ=IQ+I
10. R(IJ)=0.0
11. IF(I=J)20,15,20
12. 15 R(IJ)=1.0
13. 20 CONTINUE
14. 25 ANORM=0.0
15. DO 35 I=1,N
16. DO 35 J=1,N
17. IF(I=J)30,35,30
18. 30 IA=I+(J-J)/2
19. ANORM=ANORM+A(IA)*A(IA)
20. 35 CONTINUE
21. IF(ANORM)165,165,40
22. ANORM=1.414*SQRT(ANORM)
23. ANRMX=ANORM*RANGE/FLOAT(N)
24. IND=0
25. THR=ANORM
26. THR=THR/FLOAT(N)
27. L=1
28. M=L+1
29. MQ=(M*M-M)/2
30. LQ=(L*L-L)/2
31. LM=L+MQ
32. 62 IF(ABS(A(LM))-THR)130,65,65
33. 65 IND=1
34. LL=L+LQ
35. MM=M+MQ
36. X=0.5*(A(LL)-A(MM))
37. Y=A(LM)/SQRT(A(LM)*A(LM)+X*X)
38. IF(X)70,75,75
39. 70 Y=Y

```

```

40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.
57.
58.
59.
60.
61.
62.
63.
64.
65.
66.
67.
68.
69.
70.
71.
72.
73.
74.
75.
76.
77.
78.

75 SINX=Y/SQRT(2.0*(1.0+(SQRT(1.0-Y*Y))))
   SINX2=SINX*SINX
78 COSX=SQRT(1.0-SINX2)
   COSX2=COSX*COSX
   SINCS=8SINX*COSX
   ILQ=N*(L-1)
   IMQ=N*(M-1)
   DO 125 I=1,N
     IQ=(I*I-1)/2
     IF(I=L)80,115,80
     IF(I=M)85,115,90
80   IM=I+MQ
85   GO TO 95
90   IM=M+IQ
95   IF(I=L)100,105,105
100  IL=I+LQ
105  GO TO 110
110  IL=L+IQ
115  IF(MV=1)120,125,120
120  ILR=ILQ+I
     IMR=IMQ+I
     X=R(ILR)*COSX-R(IMR)*SINX
     R(IMR)=R(ILR)*SINX+R(IMR)*COSX
     R(ILR)=X
125  CONTINUE
     X=2.0*A(LM)*SINCS
     Y=A(LL)*COSX2+A(MM)*SINX2-X
     X=A(LL)*SINX2+A(MM)*COSX2+X
     A(LM)=A(LL)-A(MM))*SINCS+A(LM)*(COSX2-SINX2)
     A(LL)=Y
     A(MM)=X
130  IF(M=N)135,140,135
135  M=M+1
     GO TO 60
140  IF(L=(N-1))145,150,145
145  L=L+1

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500

```

79. GO TO 55
80. 150 IF(IND=1)160,155,160
81. 155 IND=0
82. GO TO 50
83. 160 IF(THR=ANRRMX)165,165,45
84. 165 IQ=N
85. DO 185 I=1,N
86. IQ=IQ+N
87. LL=I+(I*I-I)/2
88. JQ=N*(I-2)
89. DO 185 J=1,N
90. JQ=JQ+N
91. MM=J+(J*J-J)/2
92. IF(A(LL)=A(MM))170,185,185
93. 170 X=A(LL)
94. A(LL)=A(MM)
95. A(MM)=X
96. IF(MV=1)175,185,175
97. 175 DO 180 K=1,N
98. ILR=IQ+K
99. IMR=JQ+K
100. X=R(ILR)
101. R(ILR)=R(IMR)
102. 180 R(IMR)=X
103. 185 CONTINUE
104. RETURN
105. END

```



```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.

C
C
C
C
C
C
C

SUBROUTINE BGEOM(NSECT)

PROGRAM PROVIDES BEAM CROSS-SECTION GEOMETRY
AND PROPERTIES

COMMON/XSECT/NSEG(10),Y(20,10),Z(20,10),XLEN(20,10),T(20,10),
*TORCON(10),KLOS(10),ABM(10),FIXX(10),FIYY(10),FIZZ(10)
WRITE(6,200)
200 FORMAT(///30X,'BEAM CROSS-SECTION DATA'/)
      DO 10 N=1,NSECT
      READ(5,100) NSEG(N),KLOS(N),FIYY(N),FIZZ(N)
      FIXX(N)=FIYY(N)+FIZZ(N)
      TORCON(N)=0.0
      K=NSEG(N)
      IF(KLOS(N).EQ.1)K=K+1
      READ(5,110)(Y(I,N),Z(I,N),T(I,N),I=1,K)
      ABM(N)=0.
      DO 20 I=1,K
      IF(I.EQ.K) GO TO 30
      DY=Y(I+1,N)-Y(I,N)
      DZ=Z(I+1,N)-Z(I,N)
      GO TO 40
30  CONTINUE
      DY=Y(1,N)-Y(I,N)
      DZ=Z(1,N)-Z(I,N)
40  CONTINUE
      XLEN(I,N)=SQRT(DZ*DZ+DY*DY)
      ABM(N)=ABM(N)+XLEN(I,N)*T(I,N)
20  CONTINUE
      IF(KLOS(N).EQ.1)GO TO 50
      TORCON(N)=2.*T(1,N)*T(2,N)*((XLEN(1,N)-T(1,N))*2)
      C=((XLEN(2,N)-T(2,N))*2)/(XLEN(1,N)*T(1,N)+XLEN(2,N)*T(2,N)
      C=T(1,N)*T(1,N)-T(2,N)*T(2,N)
      GO TO 10
50 DO 60 I=1,K
60 TORCON(N)=TORCON(N)+XLEN(I,N)*T(I,N)**3/3.
10 CONTINUE
00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900

```

40.	DO 130 N=1,NSECT	00004000
41.	WRITE(6,210) N,NSEG(N),KLOS(N),ABM(N),FIXX(N),FIYY(N),FIZZ(N),	00
42.	1TORCON(N)	00004200
43.	210 FORMAT(/ 10X,3M N,3X,4HNSEG,3X,4HKLOS,8X,4H ABM,8X,4HFIXX,	00004300
44.	18X,4HFIYY,8X,4HFIZZ,6X,6HTORCON/6X,3I7,5E12.5/)	00004400
45.	WRITE(6,220)	00004500
46.	220 FORMAT(/39X,1HI,9X,1HY,9X,1HZ,6X,4HXLLEN,9X,1HT)	00004600
47.	K=NSEG(N)+KLOS(N)	00004700
48.	DO 120 I=1,K	00004800
49.	WRITE(6,230) I,Y(I,N),Z(I,N),XLLEN(I,N),T(I,N)	00004900
50.	230 FORMAT(38X,I2,4E10.2)	00005000
51.	120 CONTINUE	00005100
52.	130 CONTINUE	00005200
53.	100 FORMAT(2I5,2E10.4)	00005300
54.	110 FORMAT(3E10.4)	00005400
55.	RETURN	00005500
56.	END	00005600

```

1. SUBROUTINE BASME (I,NDGREE,IX,XC,YC,ZC,E,SMASS,DICOS,RMASS,PMASS,
2. *I1,STRS,NEND)
3. C
4. C CALCULATES THE MASS MATRIX OF A BEAM ELEMENT THREE DIMENSIONAL
5. COMMON/ANG/ ANGO(150),HT(150),AL(150)
6. COMMON/XSECT/NSEG(10),Y(20,10),Z(20,10),XLEN(20,10),T(20,10),
7. *TORCON(10),KLOS(10),ABM(10),FIXX(10),FIYY(10),FIZZ(10)
8. DIMENSION XC(1),YC(1),ZC(1),E(16,1),IX(9,1),SMASS(1)
9. DIMENSION DICOS(3,3,1),PMASS(4,1),RMASS(3,3,1)
10. DIMENSION XPER(6),SK(6),SM(6)
11. DIMENSION TEMP(3,3)
12. DIMENSION STRS(1)
13. N1 = IX(1,I)
14. N2 = IX(2,I)
15. MTYPE=IX(9,I)
16. C
17. C INITIALIZE STRS ARRAY
18. C
19. C
20. C
21. C
22. C
23. C
24. C
25. C
26. C
27. C
28. C
29. C
30. C
31. C
32. C
33. C
34. C
35. C
36. C
37. C
38. C
39. C

```

```

40. A=SQRT(VAX*VAX+VAY*VAY+VAZ*VAZ)
41. AX=VAX/A
42. AY=VAY/A
43. AZ=VAZ/A
44. E3X=AZ*DICOS(2,1,1)-AY*DICOS(3,1,1)
45. E3Y=AX*DICOS(3,1,1)-AZ*DICOS(1,1,1)
46. E3Z=AY*DICOS(1,1,1)-AX*DICOS(2,1,1)
47. ANORM=SQRT(E3X*E3X+E3Y*E3Y+E3Z*E3Z)
48. DICOS(1,3,1)=E3X/ANORM
49. DICOS(2,3,1)=E3Y/ANORM
50. DICOS(3,3,1)=E3Z/ANORM
51. C ESTABLISH LOCAL GLOBAL DIRECTION COSINE MATRIX
52. DICOS(1,2,1)=DICOS(2,3,1)*DICOS(3,1,1)-DICOS(3,3,1)*DICOS(2,1,1)
53. DICOS(2,2,1)=DICOS(3,3,1)*DICOS(1,1,1)-DICOS(1,3,1)*DICOS(3,1,1)
54. DICOS(3,2,1)=DICOS(1,3,1)*DICOS(2,1,1)-DICOS(2,3,1)*DICOS(1,1,1)
55. C ASSEMBLE TRANSLATIONAL MASS
56. EMASS=E(1,MTYP)*ABM(KBM)*AL(I)
57. RT=.5*EMASS
58. RX=.5*E(1,MTYP)*AL(I)*FIXX(KBM)
59. RY=EMASS*AL(I)*AL(I)/24.+5*E(1,MTYP)*AL(I)*FIYY(KBM)
60. RZ=EMASS*AL(I)*AL(I)/24.+5*E(1,MTYP)*AL(I)*FIZZ(KBM)
61. C***** COMPUTE VIBRATION PERIOD FOR BEAMS *****
62. NCR0=IX(6,I)
63. SK(1)=ABM(NCR0)*E(2,MTYP)/AL(I)
64. SK(2)=TORCON(NCR0)*E(2,MTYP)/(2.*AL(I)*(1.+E(7,MTYP)))
65. SK(3)=4.0*E(2,MTYP)*FIZZ(NCR0)/AL(I)
66. SK(4)=3.0*SK(3)/AL(I)**2
67. SK(5)=SK(3)*FIYY(NCR0)/FIZZ(NCR0)
68. SK(6)=SK(4)*FIYY(NCR0)/FIZZ(NCR0)
69. SM(1)=RT
70. SM(2)=RX
71. SM(3)=RZ
72. SM(4)=RT
73. SM(5)=RY
74. SM(6)=RT
75. PI=3.1416
76. DO 20 K=1,6
77. XPER(K)=SQRT(SM(K)/SK(K))*2.0*PI
78. 20 CONTINUE

```


79.
80.
81.
82.
83.
84.
85.
86.
87.
88.
89.
90.
91.
92.
93.
94.
95.
96.
97.
98.
99.
100.

```

WRITE(6,30) I,(XPER(J),J=1,6)
30 FORMAT(6X,I5,3X,6E10,3)
DO 4 N = 3,4
  KK=IX(N,I)
  NN=(KK-1)*NDGREE
  DO 5 M = 1,3
    5 SMASS(NN+M) = RT + SMASS(NN+M)
  4 CONTINUE
C
  ASSEMBLE ROTATIONAL MASS IN GLOBAL COORDINATES
  DO 50 II=1,3
    DO 50 JJ=1,3
      TEMP(II,JJ)=DICOS(II,1,I)*DICOS(JJ,1,I)*RX+
      * DICOS(II,2,I)*DICOS(JJ,2,I)*RY+DICOS(II,3,I)*DZ
    50 CONTINUE
    DO 60 M=3,4
      NN=IX(M,I)
      DO 70 II=1,3
        DO 70 JJ=1,3
          70 RMASS(II,JJ,N)=RMASS(II,JJ,N)+TEMP(II,JJ)
        60 CONTINUE
      RETURN
    END

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000

```

1. SUBROUTINE TASME(JE,NDGREE,IX,XC,YC,ZC,E,SMASS,SIDE,IP,STRS,
2. *NTOT,I1,DICOS,RMASS)
3. DIMENSION XC(1),YC(1),ZC(1),E(16,1),IX(9,1),SMASS(1),SIDE(1),IP(1)
4. DIMENSION X(3),Y(3),Z(3),XXI(3),YYI(3),ZZI(3)
5. DIMENSION VA(3),VB(3)
6. DIMENSION DICOS(3,3),RMASS(3,3,1)
7. DIMENSION STRS(1)
8. DATA LOC/3/,NDAT/7/
9. DO 105 I=1,3
10. J=IP(I+1)
11. K=IX(I,JE)
12. L=IX(J,JE)
13. X(I)=XC(L)=XC(K)
14. Y(I)=YC(L)=YC(K)
15. Z(I)=ZC(L)=ZC(K)
16. XXI(I)=0.0
17. YYI(I)=0.0
18. ZZI(I)=0.0
19. 105 SIDE(I)=SQRT(X(I)**2+Y(I)**2+Z(I)**2)
20. AREA=0.5*(X(1)*(Y(2)-Z(2))+Y(1)*(Z(2)-X(2))+Z(1)*(X(2)-Y(2)))
21. A=AREA/6.
22. IF(AREA.LE.0.) GO TO 115
23. MTYP=IX(9,JE)
24. C
25. C
26. C
27. C
28. C
29. INITIALIZE STRS STORAGE
30. NTOT=LOC*NDAT*I1
31. DO 200 I=1,NTOT
32. STRS(I)=0.
33. N=LOC*I1
34. DO 210 I=1,N
35. J=NDAT*I
36. STRS(J)=E(3,MTYP)
37. C
38. C
39. C
40. C
41. CALCULATE ELEMENT MASS
42. C
43. C
44. C
45. C
46. C
47. C
48. C
49. C
50. C
51. C
52. C
53. C
54. C
55. C
56. C
57. C
58. C
59. C
60. C
61. C
62. C
63. C
64. C
65. C
66. C
67. C
68. C
69. C
70. C
71. C
72. C
73. C
74. C
75. C
76. C
77. C
78. C
79. C
80. C
81. C
82. C
83. C
84. C
85. C
86. C
87. C
88. C
89. C
90. C
91. C
92. C
93. C
94. C
95. C
96. C
97. C
98. C
99. C
100. C
101. C
102. C
103. C
104. C
105. C
106. C
107. C
108. C
109. C
110. C
111. C
112. C
113. C
114. C
115. C
116. C
117. C
118. C
119. C
120. C
121. C
122. C
123. C
124. C
125. C
126. C
127. C
128. C
129. C
130. C
131. C
132. C
133. C
134. C
135. C
136. C
137. C
138. C
139. C
140. C
141. C
142. C
143. C
144. C
145. C
146. C
147. C
148. C
149. C
150. C
151. C
152. C
153. C
154. C
155. C
156. C
157. C
158. C
159. C
160. C
161. C
162. C
163. C
164. C
165. C
166. C
167. C
168. C
169. C
170. C
171. C
172. C
173. C
174. C
175. C
176. C
177. C
178. C
179. C
180. C
181. C
182. C
183. C
184. C
185. C
186. C
187. C
188. C
189. C
190. C
191. C
192. C
193. C
194. C
195. C
196. C
197. C
198. C
199. C
200. C
201. C
202. C
203. C
204. C
205. C
206. C
207. C
208. C
209. C
210. C
211. C
212. C
213. C
214. C
215. C
216. C
217. C
218. C
219. C
220. C
221. C
222. C
223. C
224. C
225. C
226. C
227. C
228. C
229. C
230. C
231. C
232. C
233. C
234. C
235. C
236. C
237. C
238. C
239. C
240. C
241. C
242. C
243. C
244. C
245. C
246. C
247. C
248. C
249. C
250. C
251. C
252. C
253. C
254. C
255. C
256. C
257. C
258. C
259. C
260. C
261. C
262. C
263. C
264. C
265. C
266. C
267. C
268. C
269. C
270. C
271. C
272. C
273. C
274. C
275. C
276. C
277. C
278. C
279. C
280. C
281. C
282. C
283. C
284. C
285. C
286. C
287. C
288. C
289. C
290. C
291. C
292. C
293. C
294. C
295. C
296. C
297. C
298. C
299. C
300. C
301. C
302. C
303. C
304. C
305. C
306. C
307. C
308. C
309. C
310. C
311. C
312. C
313. C
314. C
315. C
316. C
317. C
318. C
319. C
320. C
321. C
322. C
323. C
324. C
325. C
326. C
327. C
328. C
329. C
330. C
331. C
332. C
333. C
334. C
335. C
336. C
337. C
338. C
339. C
340. C
341. C
342. C
343. C
344. C
345. C
346. C
347. C
348. C
349. C
350. C
351. C
352. C
353. C
354. C
355. C
356. C
357. C
358. C
359. C
360. C
361. C
362. C
363. C
364. C
365. C
366. C
367. C
368. C
369. C
370. C
371. C
372. C
373. C
374. C
375. C
376. C
377. C
378. C
379. C
380. C
381. C
382. C
383. C
384. C
385. C
386. C
387. C
388. C
389. C
390. C
391. C
392. C
393. C
394. C
395. C
396. C
397. C
398. C
399. C
400. C
401. C
402. C
403. C
404. C
405. C
406. C
407. C
408. C
409. C
410. C
411. C
412. C
413. C
414. C
415. C
416. C
417. C
418. C
419. C
420. C
421. C
422. C
423. C
424. C
425. C
426. C
427. C
428. C
429. C
430. C
431. C
432. C
433. C
434. C
435. C
436. C
437. C
438. C
439. C
440. C
441. C
442. C
443. C
444. C
445. C
446. C
447. C
448. C
449. C
450. C
451. C
452. C
453. C
454. C
455. C
456. C
457. C
458. C
459. C
460. C
461. C
462. C
463. C
464. C
465. C
466. C
467. C
468. C
469. C
470. C
471. C
472. C
473. C
474. C
475. C
476. C
477. C
478. C
479. C
480. C
481. C
482. C
483. C
484. C
485. C
486. C
487. C
488. C
489. C
490. C
491. C
492. C
493. C
494. C
495. C
496. C
497. C
498. C
499. C
500. C

```


40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.
57.
58.
59.
60.
61.
62.
63.
64.
65.
66.
67.
68.
69.
70.
71.
72.
73.
74.
75.
76.
77.
78.

C

EMASS=AREA*E(1,MTYP)*E(5,MTYP)/3.
RM=E(1,MTYP)*E(5,MTYP)/12.

DO 110 N=1,3

J=(IX(N,JE)-1)*NDGREE

DO 110 I=1,3

110 SMASS(I+J)=8MASS(I+J)+EMASS

N1=IX(1,JE)

N2=IX(2,JE)

N3=IX(3,JE)

XCEN=(XC(N1)+XC(N2)+XC(N3))/3.

YCEN=(YC(N1)+YC(N2)+YC(N3))/3.

DO 120 I=1,3

J=IP(I+1)

M=IP(I+2)

K=IX(I,JE)

L=IX(J,JE)

N=IX(M,JE)

XBAR=XCEN-XC(K)

YBAR=YCEN-YC(K)

XCORD=0.5*(XC(L)-XC(K))

YCORD=0.5*(YC(L)-YC(K))

XKORD=0.5*(XC(N)-XC(K))

YKORD=0.5*(YC(N)-YC(K))

CALL QUAD(0.,0.,XCORD,YCORD,XBAR,YBAR,A,XXI(I),YYI(I))

CALL QUAD(XBAR,YBAR,XKORD,YKORD,0.,0.,A,XXI(I),YYI(I))

120 CONTINUE

DO 125 J=1,3

XXI(J)=E(1,MTYP)*E(5,MTYP)*XXI(J)

YYI(J)=E(1,MTYP)*E(5,MTYP)*YYI(J)

125 ZZI(J)=XXI(J)+YYI(J)

C

C

C

C

C

CONSTRUCT INITIAL ELEMENT NORMAL

VA(1)=X(1)

VA(2)=Y(1)

VA(3)=Z(1)

00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800

```

79. VB(1)=X(3)
80. VB(2)=Y(3)
81. VB(3)=Z(3)
82. CALL CROSS(VA,VB,DICOS(1,3),ANG,PMAG,0)
83. PMAG=0.
84. DO 140 I=1,3
85. PMAG=PMAG+((VA(I)-VB(I))**2)
86. PMAG=SQRT(PMAG)
87. DO 145 I=1,3
88. DICOS(I,1)=(VA(I)-VB(I))/PMAG
89. CALL CROSS(DICOS(1,3),DICOS(1,1),DICOS(1,2),ANG,PMAG,0)
90. DO 160 II=1,3
91. N=IX(II,JE)
92. DO 160 I=1,3
93. DO 160 J=1,3
94. RMASS(I,J,N)=RMASS(I,J,N)+DICOS(I,1)*DICOS(J,1)*XXI(II)
95. **DICOS(I,2)*DICOS(J,2)*YYI(II)+DICOS(I,3)*DICOS(J,3)*ZZI(II)
96.
97. 160 CONTINUE
98. RETURN
99. 115 WRITE(6,1000) AREA,JE
100. 1000 FORMAT(1H0,5X,'AREA=1,E10.4,'OF ELEMENT',I5)
101. STOP
END
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100

```

```

1. SUBROUTINE FRCIN(NUMEL,NUMNP,NDGREE,IX,E,XO,FINT,OLDS)
2. COMMON/DYNAM/DEL,TIMEND,MXSTEP,NTSTEP
3. COMMON/JUNK/TL(9,300),POINT(3,3,300)
4. COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6),
5. *STRS(10000)
6. COMMON /BEAMS/RMASS(3,3,100),DICOS(3,3,150),BAR(1200),
7. *PMASS(4,100),NSECT
8. COMMON XC(300),YC(300),ZC(300),UD(900),UD1(900),UD2(900)
9. DIMENSION IX(9,1),E(16,1),XO(1),B1(3),B3(3),TEMP1(3),TEMP2(3)
10. DIMENSION TB1(3),TB3(3)
11. DIMENSION FINT(1)
12. DIMENSION OLDS(1)
13. DOUBLE PRECISION B1,B3,B31S,B32S,XL
14. MEQ=NDGREE*NUMNP
15. DO 15 I=1,MEQ
16. FINT(I)=0.
17. C1=DEL*DELT/2.
18.
19. C CALCULATE B3 VECTOR IN RIGID BODY COORDINATES
20. C
21. NRN=NUMNP
22. DO 10 I=1,NRN
23. M=NDGREE*I
24. B3(1)=DELT*UD1(M=1)+C1*(UD1(M=2)*UD1(M)+UD2(M=1))
25. B3(2)=DELT*UD1(M=2)+C1*(UD1(M=1)*UD1(M)+UD2(M=2))
26. B1(2)=DELT*UD1(M)+C1*(UD1(M=2)*UD1(M=1)+UD2(M))
27. B31S=B3(1)**2
28. B32S=B3(2)**2
29. B3(3)=0.5*(B31S+B32S)
30. XL=DSQRT(B31S+B32S+(1.+B3(3))**2)
31. B3(1)=B3(1)/XL
32. B3(2)=B3(2)/XL
33. B3(3)=(1.+B3(3))/XL
34. B1(3)=(-B3(1)-B1(2)*B3(2))/B3(3)
35. B1(1)=DSQRT(1.-B1(2)**2-B1(3)**2)
36.
37. C TRANSFORM VECTOR FROM RIGID BODY TO GLOBAL COORDINATES
38. C
39. DO 200 KK=1,3

```

```

40. TB1(KK)=B1(KK)
41. TB3(KK)=B3(KK)
42. CALL GMPRD(TL(1,I),TB3,TEMP1,3,3,1)
43. CALL GMPRD(TL(1,I),TB1,TEMP2,3,3,1)
44.
45. C UPDATE LAMBDA TRANSFORMATION MATRIX
46. C
47. TL(1,I)=TEMP2(1)
48. TL(2,I)=TEMP2(2)
49. TL(3,I)=TEMP2(3)
50. CALL CROSS(TEMP1,TEMP2,TL(4,I),BETA,CMAG,1)
51. TL(7,I)=TEMP1(1)
52. TL(8,I)=TEMP1(2)
53. TL(9,I)=TEMP1(3)
54. 10 CONTINUE
55. HJ=0
56. DO 20 JE=1,NUMEL
57. MYP = IX(9,JE)
58. KEL=IX(8,JE)
59. IND = INDEX(JE)
60. IF(KEL.GT.1) GO TO 100
61. CALL TPRCIN(JE,NDGREE,XC,YC,ZC,IX,E(1,MTYP),STRS(IND),STRESS,
62. *STRAIN,XO,FINT,IP,I1,SIDE,OLDS)
63. GO TO 110
64. 100 CONTINUE
65. MJ=MJ+1
66. I=IX(3,JE)
67. J=IX(4,JE)
68. ME(MJ-1)*24+1
69. M1=M+12
70. CALL BFRICIN(JE,NDGREE,XC,YC,ZC,IX,E,XO,FINT,STRS,STRAIN,STRESS,
71. *I1,INDEX,SMASS,TL(1,I),TL(1,J),BAR(M),BAR(M1))
72. 110 CONTINUE
73. 20 CONTINUE
74. RETURN
75. END

```

```

00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500

```



```

1. SUBROUTINE TFRGIN(JE,ND,XC,YC,ZC,NODE,E,STRS,STRESS,STRAIN,UD, 00000100
2. *FINT,IP,IT,SIDE,OLDS) 00000200
3. COMMON/JUNK/TL(9,300),POINT(3,3,300) 00000300
4. COMMON/DYNAM/DEL 00000400
5. DIMENSION XC(1),YC(1),ZC(1),NODE(9,1),E(1),STRS(1),STRESS(1), 00000500
6. *STRAIN(1),UD(1),IP(1),SIDE(1) 00000600
7. DIMENSION FORCE(16),X(3),Y(3),Z(3),UX(3),UY(3),UZ(3),D(3),EPS(3), 00000700
8. 1SF(3),T(3) 00000800
9. DIMENSION STRNM(3),XROT(3),YROT(3),FZ(3),SMX(3),SMY(3) 00000900
10. DIMENSION CURV(3,3),EF(3,3),EM(3,3) 00001000
11. DIMENSION FINT(1) 00001100
12. DIMENSION OLDS(1) 00001200
13. DIMENSION TEMP(3) 00001300
14. ROTATIONS AT EACH NODE 00001400
15. CALL VECTOR(XC,YC,ZC,UD,NODE,JE,D,S,IP,A,B,C,ABC) 00001500
16. C1=1.0/(D(1)*D(3)) 00001600
17. CX=SQRT(8*(S=D(2))*C1) 00001700
18. CY=SQRT((S=D(1))*(S=D(3))*C1) 00001800
19. AREA=D(1)*D(3)*CX*CY 00001900
20. THETA=ACOS(CX) 00002000
21. CALL VECT(THETA,CXX,CYX,CZX) 00002100
22. THETA=1.570796+THETA 00002200
23. CALL VECT(THETA,CXY,CYY,CZY) 00002300
24. DO 100 I=1,3 00002400
25. NOD=NODE(I,JE) 00002500
26. K=NOD 00002600
27. CALL GMPRD(TL(1,K),POINT(1,I,K),T,3,3,1) 00002700
28. XROT(I)=CXY*T(1)+CYY*T(2)+CZY*T(3) 00002800
29. YROT(I)=CXX*T(1)-CXY*T(2)-CZY*T(3) 00002900
30. 100 00003000
31. STRAIN OF ELEMENT SIDES 00003100
32. DO 105 I=1,3 00003200
33. J=IP(I+1) 00003300
34. I1=NODE(I,JE) 00003400
35. I2=NODE(J,JE) 00003500
36. X(I)=XC(I2)-XC(I1) 00003600
37. Y(I)=YC(I2)-YC(I1) 00003700
38. Z(I)=ZC(I2)-ZC(I1) 00003800
39. L1=ND*(I1=1) 00003900

```

```

40.      L2=ND*(I2=1)
41.      UX(I)=UD(L2+1)-UD(L1+1)
42.      UY(I)=UD(L2+2)-UD(L1+2)
43.      UZ(I)=UD(L2+3)-UD(L1+3)
44.      DO 110 I=1,3
45.      DS=SIDE(3*JE+I=3)
46.      EPS(I)=(2.*(X(I)*UX(I)+Y(I)*UY(I)+Z(I)*UZ(I))+UX(I)**2+UY(I)**2
47.      1+UZ(I)**2)/((D(I)+DS)*DS)
48.      C      MEMBRANE STRAIN
49.      D1=C1/(4.*CX*CX)
50.      D2=C1/(4.*CY*CY)
51.      D3=0.5/(CX*CY)
52.      STRNM(1)=(D(1)*(D(1)+D(3))*EPS(1)-D(2)**2*EPS(2)
53.      1+D(3)*(D(1)+D(3))*EPS(3))*D1
54.      STRNM(2)=(D(1)*(D(3)-D(1))*EPS(1)+D(2)**2*EPS(2)
55.      1+D(3)*(D(1)-D(3))*EPS(3))*D2
56.      STRNM(3)=(EPS(1)-EPS(3))*D3
57.      C      CURVATURES
58.      CALL CURVAT(CURV,AREA,XROT,YROT,D,CX,CY)
59.      C      STRAINS AND STRESSES ACROSS THICKNESS
60.      DO 115 M = 1,3
61.      IND=7*IT*(M=1)+1
62.      H = E(5)
63.      CALL GENFOR (STRNM,CURV(1,M),EM(1,M),EF(1,M),E,H,IT,STRS(IND))
64.      115 CONTINUE
65.      SF1=0.
66.      SF2=0.
67.      SF3=0.
68.      DO 135 I=1,3
69.      SF1=SF1+((D(1)+D(3))*D1*EF(1,I)-(D(1)-D(3))*D2*EF(2,I)
70.      1+D3/D(1)*EF(3,I))/D(1)
71.      SF2=SF2+((-D1*EF(1,I)+D2*EF(2,I))
72.      135 SF3=SF3+((D(1)+D(3))*D1*EF(1,I)+(D(1)-D(3))*D2*EF(2,I)
73.      1-D3/D(3)*EF(3,I))/D(3)
74.      CONST=AREA/3.
75.      SF(1)=CONST*SF1
76.      SF(2)=CONST*SF2
77.      SF(3)=CONST*SF3
78.      C

```



```

79.
80.
81.
82.
83.
84.
85.
86.
87.
88.
89.
90.
91.
92.
93.
94.
95.
96.
97.
98.
99.
100.
101.
102.
103.
104.
105.
106.
107.
108.

C ADDITION OF MEMBRANE DAMPING (VISCOUS) FORCES
C
  ID=3*(JE-1)
  DO 150 I=1,3
    EPSS=OLDS(ID+I)
    DEPS=EPS(I)-EPSS
    OLDS(ID+I)=EPS(I)
    SF(I)=SF(I)+2.*E(8)*AREA*(DEPS/DELTA)*SQRT(E(1)*E(2))*E(5)
  150 CONTINUE

C NODAL FORCES AND MOMENTS
CALL NODALF(EM,SMX,SMY,FZ,AREA)
DO 140 I=1,3
  J=IP(I+2)
  K=ND*(I-1)
  N=NODE(I,JE)
  FORCE(K+1)=-(X(I)+UX(I))*SF(I)+(X(J)+UX(J))*SF(J)+A/ABC*FZ(I)
  FORCE(K+2)=-(Y(I)+UY(I))*SF(I)+(Y(J)+UY(J))*SF(J)+B/ABC*FZ(I)
  FORCE(K+3)=-(Z(I)+UZ(I))*SF(I)+(Z(J)+UZ(J))*SF(J)+C/ABC*FZ(I)
  FORCE(K+4)=CMX*SMX(I)+CXY*SMY(I)
  FORCE(K+5)=CYX*SMX(I)+CYY*SMY(I)
  FORCE(K+6)=CZX*SMX(I)+CZY*SMY(I)
  CALL GTPRD(TL(1,N),FORCE(K+4),TEMP,3,3,1)
DO 145 KK=1,3
  LOC=ND*(N-1)+KK
  FINT(LOC)=FINT(LOC)+FORCE(K+KK)
  FINT(LOC+3)=FINT(LOC+3)+TEMP(KK)
145 CONTINUE
140 CONTINUE
  RETURN
END
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800

```

```

1. SUBROUTINE BFCIN (L,ND,XC,YC,ZC,IX,E,UD,FINT,STRS,STRAIN,STRESS, 00000100
2. I1,INDEX,SMASS,EUI,EUJ,TRAI,TRAJ) 00000200
3. ROUTINE CALCULATES INTERNAL FORCES FOR THREE-D BEAM 00000300
4. TRAI AND TRAJ CONTAIN I BAR 0 ,J BAR 0 AND J HAT 0 FOR NODES I + J 00000400
5. DIMENSION FINT(1) 00000500
6. COMMON/ANG/ ANGO(150),HT(150),AL(150) 00000600
7. DIMENSION XC(1),YC(1),IX(09,1),UD(1),E(16,1) 00000700
8. ,STRAIN(1),SMASS(1),STRESS(1),STRS(1),INDEX(1) 00000800
9. ,ZC(1),EUI(1),EUJ(1),EH(9),TRAI(1),TRAJ(1) 00000900
10. ,EJBI(3),EJB(3) 00001000
11. ,TEMP1(3),TEMP2(3),TEMP3(3),TEMP4(3),TEMP5(3),OMEGI(3,3), 00001100
12. OMEGJ(3,3),DISP(2),ROT(6),AFORCE(6),BMOMT(6) 00001200
13. I2=I1*2 00001300
14. N1 = IX(1,L) 00001400
15. N2 = IX(2,L) 00001500
16. N3=IX(3,L) 00001600
17. N4=IX(4,L) 00001700
18. XX=XC(N2)-XC(N1) 00001800
19. YY= YC(N2)-YC(N1) 00001900
20. ZZ = ZC(N2)-ZC(N1) 00002000
21. N1N=(N1-1)*ND 00002100
22. N2N=(N2-1)*ND 00002200
23. N3N=(N3-1)*ND 00002300
24. N4N=(N4-1)*ND 00002400
25. II = N1 00002500
26. JJ = N2 00002600
27. OMEGI(1,1)=0. 00002700
28. OMEGI(1,2)=TRAI(12) 00002800
29. OMEGI(1,3)=TRAJ(11) 00002900
30. OMEGI(2,1)=TRAJ(12) 00003000
31. OMEGI(2,2)=0. 00003100
32. OMEGI(2,3)=TRAI(10) 00003200
33. OMEGI(3,1)=TRAI(11) 00003300
34. OMEGI(3,2)=TRAJ(10) 00003400
35. OMEGI(3,3)=0. 00003500
36. OMEGJ(1,1)=0. 00003600
37. OMEGJ(1,2)=TRAJ(12) 00003700
38. OMEGJ(1,3)=TRAJ(11) 00003800
39. OMEGJ(2,1)=TRAJ(12) 00003900

```

```

40. OMEGJ(2,2)=0.
41. OMEGJ(2,3)=TRAJ(10)
42. OMEGJ(3,1)=TRAJ(11)
43. OMEGJ(3,2)=TRAJ(10)
44. OMEGJ(3,3)=0.
45. TEMP1(1)=TRAI(10)
46. TEMP1(2)=TRAI(11)
47. TEMP1(3)=TRAI(12)
48. TEMP2(1)=TRAJ(10)
49. TEMP2(2)=TRAJ(11)
50. TEMP2(3)=TRAJ(12)
51. CALL GMPRD(EUI,TEMP1,TEMP3,3,3,1)
52. CALL GMPRD(EUJ,TEMP2,TEMP4,3,3,1)
53. DIX=UD(N3N+1)+TEMP3(1)-TRAI(7)
54. DJX=UD(N4N+1)+TEMP4(1)-TRAJ(7)
55. DIY=UD(N3N+2)+TEMP3(2)-TRAI(8)
56. DJY=UD(N4N+2)+TEMP4(2)-TRAJ(8)
57. DIZ=UD(N3N+3)+TEMP3(3)-TRAI(9)
58. DJZ=UD(N4N+3)+TEMP4(3)-TRAJ(9)
59. UD(N1N+1)=DIX
60. UD(N1N+2)=DIY
61. UD(N1N+3)=DIZ
62. UD(N2N+1)=DJX
63. UD(N2N+2)=DJY
64. UD(N2N+3)=DJZ
65.
66. FIND RIDID BODY ROTATION AFTER DEFORMATION W/T ORIGINAL COORDINAT
67. FIND STRAINS AT NODES, HT IS THICKNESS OF BEAM ELEMENT
68. DIJX=DJX-DIX
69. DIJY=DJY-DIY
70. DIJZ = DJZ - DIZ
71. DX=XX+DIJX
72. DY=YY+DIJY
73. DZ = ZZ + DIJZ
74. AL2 = AL(L)*AL(L)
75. FAC = CENTROIDAL AXIS STRAIN
76. FAC = 2.*(XX*DIJX+YY*DIJY+ZZ*DIJZ)+DIJX*DIJX+DIJY*DIJY+DIJZ*DIJZ
77. ALN2=AL2+FAC
78. ALN=SQRT(ALN2)

```

```

79.      EH(1) = DX/ALN
80.      EH(2) = DY/ALN
81.      EH(3) = DZ/ALN
82.      DO 50 I=1,3
83.      I4=3 + I
84.      I3=6 + I
85.      EJB(I)=EUI(I)*TRAI(I)+EUI(I4)*TRAI(5)+EUI(I3)*TRAI(6)
86.      EJB(I)=EUI(I)*TRAJ(I)+EUI(I4)*TRAJ(5)+EUI(I3)*TRAJ(6)
87.      DO 51 I=1,3
88.      I4=3 + I
89.      EH(I4)=(EJB(I) + EJB(I))/2.
90.      CALL CROSS(EH(1),EH(4),EH(7),DUMA,DUMB,0)
91.      CALL CROSS(EH(7),EH(1),EH(4),DUMA,DUMB,0)
92.      C DETERMINE DEFORMATION ROTATIONS OF BEAM
93.      C EH CONTAINS MU, EUI + EUJ CONTAIN LAMDA I + LAMDA J
94.      C CALCULATE DEFORMATION ROTATIONS LOCAL COORDINATES(PHIX,PHIZ ECT.)
95.      CALL CROSS(EJB,I,EJB,I,TEMP3,ROT(1),DUMA,1)
96.      CALL GTPRD(EH,TEMP3,TEMP4,3,3,0)
97.      ROT(1)=TEMP4(1)
98.      CALL GMPRD(EUI,TRAI,TEMP1,3,3,1)
99.      CALL GTPRD(EH,TEMP1,TEMP2,3,3,1)
100.      ROT(3)=TEMP2(3)
101.      ROT(5)=TEMP2(2)
102.      CALL GMPRD(EUJ,TRAJ,TEMP1,3,3,1)
103.      CALL GTPRD(EH,TEMP1,TEMP2,3,3,1)
104.      ROT(4)=TEMP2(3)
105.      ROT(6)=TEMP2(2)
106.      C DETERMINE LOCAL INTERNAL FORCES
107.      C
108.      STREH = FAC/(AL(L)+ALN)/AL(L)
109.      MTYP = IX(9,L)
110.      DISP(1)=ALN
111.      DISP(2)=STREH
112.      NOPT=3
113.      CALL LOCFRC(L,DISP,ROT,AFORCE,BMOMT,E,INDEX,I1,STRAIN,
114.      *STRESS,STRS,SMASS,IX,AL(L),MTYP,NOPT,NUMEL)
115.      C ACCUMULATE INTERNAL ELEMENTAL FORCES TO FINT
116.      C COMPUTE GLOBAL COMPONENTS OF FINT
117.      TEMP1(1)=AFORCE(1)

```


00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013500
00013600
00013700
00013800
00013900
00014000
00014100
00014200
00014300
00014400
00014500
00014600
00014700
00014800

```

TEMP1(2)=AFORCE(3)
TEMP1(3)=AFORCE(5)
CALL GMPRD(EH,TEMP1,TEMP2,3,3,1)
TEMP3(1)=AFORCE(2)
TEMP3(2)=AFORCE(4)
TEMP3(3)=AFORCE(6)
CALL GMPRD(EH,TEMP3,TEMP1,3,3,1)
DO 23 I=1,3
  FINT(N3N+I)=FINT(N3N+I) + TEMP2(I)
  FINT(N4N+I)=FINT(N4N+I) + TEMP1(I)
  CALL GTPRD(EUI,TEMP2,TEMP3,3,3,1)
  CALL GTPRD(OMEGI,TEMP3,TEMP2,3,3,1)
  CALL GTPRD(EUJ,TEMP1,TEMP3,3,3,1)
  CALL GTPRD(OMEGJ,TEMP3,TEMP1,3,3,1)
  TEMP4(1)=BMOMT(1)
  TEMP4(2)=BMOMT(3)
  TEMP4(3)=BMOMT(5)
  TEMP5(1)=BMOMT(2)
  TEMP5(2)=BMOMT(4)
  TEMP5(3)=BMOMT(6)
  CALL GMPRD(EH,TEMP4,TEMP3,3,3,1)
  CALL GTPRD(EUI,TEMP3,TEMP4,3,3,1)
  CALL GMPRD(EH,TEMP5,TEMP3,3,3,1)
  CALL GTPRD(EUJ,TEMP3,TEMP5,3,3,1)
  I=0
DO 24 J=4,6
  I=I+1
  FINT(N3N+J)=FINT(N3N+J) + TEMP4(I) + TEMP2(I)
  FINT(N4N+J)=FINT(N4N+J) + TEMP5(I) + TEMP1(I)
24 RETURN
END

```

118.
119.
120.
121.
122.
123.
124.
125.
126.
127.
128.
129.
130.
131.
132.
133.
134.
135.
136.
137.
138.
139.
140.
141.
142.
143.
144.
145.
146.
147.
148.

```

1. SUBROUTINE LOCFRC(JE, DISP, ROT, AFORCE, BMOMT, E, INDEX, I1, STRAIN,
2. *STRESS, STRS, SMASS, IX, EL, MTYP, NOPT, NUMEL)
3. COMMON/DYNAM/DELT, TIMEND, MXSTEP, N1STEP, TIME, PCDAMP
4. COMMON/XSECT/NSEG(10), Y(20,10), Z(20,10), XLEN(20,10), T(20,10),
5. *TORCON(10), KLOS(10), ABM(10), FIXX(10), FIYY(10), FIZZ(10)
6.
7. THIS ROUTINE CALCULATES THE LOCAL INTERNAL FORCES FOR +
8. NOPT = 1 3-D AXIAL SPRING ELEMENT
9. NOPT = 2 3-D LINEAR ELASTIC BEAM ELEMENT
10. NOPT = 3 3-D ELASTO-PLASTIC BEAM ELEMENT
11.
12. DIMENSION DISP(1), ROT(1), AFORCE(1), BMOMT(1), E(16,1), INDEX(1),
13. * STRAIN(1), EPSOLD(150), SMASS(1), IX(09,1), STRESS(1), STRS(1)
14.
15. IF (TIME.GT.0.0) GO TO 1
16. EPSOLD(JE)=0.
17. 1 IF (NOPT.EQ.3) GO TO 3
18.
19. CALCULATE DAMPING FORCE FOR SPRING AND ELASTIC BEAM
20.
21. FDAMP=0.
22. PCDAMP=E(12,MTYP)
23. IF (PCDAMP) 12,12,11
24. 11 N3=IX(3,JE)
25. N4=IX(4,JE)
26. N3N=(N3-1)*6+1
27. N4N=(N4-1)*6+1
28. TMASS=(SMASS(N3N)+SMASS(N4N))/2.
29. DDOT=(DISP(2)-EPSOLD(JE))*DISP(1)/DELT
30. STIFF=E(2,MTYP)*E(7,MTYP)/EL
31. FDAMP=2.*PCDAMP*DDOT*SQRT(TMASS*STIFF)
32. EPSOLD(JE)=DISP(2)
33. 12 CONTINUE
34.
35. IF (NOPT.EQ.1) GO TO 13
36. DELSTR=DISP(2)
37. GO TO 17
38.
39. SPRING ELEMENT

```

```

00000100
00000200
00000300
00
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900

```



```

40. C
41. C
42. C
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.
57.
58.
59.
60.
61.
62.
63.
64.
65.
66.
67.
68.
69.
70.
71.
72.
73.
74.
75.
76.
77.
78.

TORC IS TENSION OR COMPRESSION CONTROL FOR SPRING ELEMENTS

13 TORC=E(1,MTYP)
   SLACK=E(4,MTYP)
   DELSTR=DISP(2)-SLACK
   IF(TORC)15,17,16
15 IF(DELSTR.GE.0.)DELSTR=0.
   GO TO 17
16 IF(DELSTR.LE.0.)DELSTR=0.
17 AFORCE(2)=E(2,MTYP)*DELSTR*E(7,MTYP)+FDAMP
   AFORCE(1)=AFORCE(2)
   GO TO 4

ELASTO=PLASTIC BEAMS

3 CONTINUE

      BEAM= CROSS=SECTION FORCES

LEFT END

KBM=IX(6,JE)
NPT=NSEG(KBM)+KLOS(KBM)
IND=INDEX(JE)
EXT=DISP(2)
TWIST=ROT(1)/EL
CURVZ=2.*(2.*ROT(5)+ROT(6))/EL
CURVY=2.*(2.*ROT(3)+ROT(4))/EL
CALL GENFBM(JE,E(1,MTYP),STRESS,STRAIN,STRS(IND),IM,EXT,CURVZ,
*CURVY,TWIST,FX,AMZ,AMY,AMX)
BMOMT(1)=AMX
BMOMT(3)=AMY
BMOMT(5)=AMZ
AFORCE(1)=FX

RIGHT END

IND=IND+4*NP1

```

79.
80.
81.
82.
83.
84.
85.
86.
87.
88.
89.
90.
91.
92.
93.

```

CURVZ=2.*(ROT(5)+2.*ROT(6))/EL
CURVY=-2.*(ROT(3)+2.*ROT(4))/EL
CALL GENFBM(JE,E(1,MTYP),STRESS,STRAIN,STRS(IND),IX,EXT,CURVZ,
*CURVY,TWIST,FX,AMZ,AMY,AMX)
  BMOMT(2)=AMX
  BMOMT(4)=AMY
  BMOMT(6)=AMZ
  AFORCE(2)=FX
  AFORCE(4)=-(BMOMT(5)+BMOMT(6))/EL
  AFORCE(3)=AFORCE(4)
  AFORCE(6)=-(BMOMT(3)+BMOMT(4))/EL
  AFORCE(5)=AFORCE(6)
    4 CONTINUE
    10 RETURN
      END

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300

```

1. SUBROUTINE GENFBM(JE,E,STRESS,STRAIN,STRS,IX,EXT,CURVZ ,CURVY,
2. *ROT,FX,AMZ,AMY,AMX)
3. DIMENSION E(1),STRESS(1),STRAIN(1),STRS(1),IX(9,1)
4. COMMON/XSECT/NSEG(10),Y(20,10),Z(20,10),XLEN(20,10),T(20,10),
5. *TORCON(10),KLOS(10),ABM(10),FIXX(10),FIYY(10),FIZZ(10)
6.
7.
8.
9.
10.
11.
12.
13.
14.
15.
16.
17.
18.
19.
20.
21.
22.
23.
24.
25.
26.
27.
28.
29.
30.
31.
32.
33.
34.
35.
36.
37.
38.
39.

```

C C C C C C C C C

```

      MOMENT CURVATURE RELATION BY PIECEWISE LINEAR INTEGRATION
      OF STRESSES OVER PLATEWORK TYPE BEAM CROSS-SECTIONS

      SIGN CONVENTION
      MZ=E*IZ*CURVZ
      MY=E*IY*CURVY

      KBM=IX(6,JE)
      IF(KBM.EQ.0) KBM=1
      KPTS=NSEG(KBM)
      IF(KLOS(KBM).EQ.1) KPTS=KPTS+1
      DO 10 I=1,KPTS
         YS=Y(I,KBM)
         ZS=Z(I,KBM)
10      STRAIN(I)=EXT-YS*CURVZ-ZS*CURVY
         DO 20 I=1,KPTS
            IND=4*(I-1)+1
20      CALL STRES(STRAIN(I),STRESS(I),E,STRS(IND))
         FX=E.
         AMZ=E.
         AMY=E.
         DO 30 I=1,KPTS
            CON=T(I,KBM)*XLEN(I,KBM)
            YS=Y(I,KBM)
            ZS=Z(I,KBM)
            IF(I.EQ.KPTS) GO TO 40
            YSN=Y(I+1,KBM)
            ZSN=Z(I+1,KBM)
            FX=FX+.5*(STRESS(I)+STRESS(I+1))*CON
            AMY=AMY+(STRESS(I)*(ZSN+2.*ZS)+STRESS(I+1)*(ZS+2.*ZSN))*CON/6.
            AMZ=AMZ+(STRESS(I)*(YSN+2.*YS)+STRESS(I+1)*(YS+2.*YSN))*CON/6.
30
31
32
33
34
35
36
37
38
39

```

00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000

```

40  GO TO 30
40  YSN=Z(1,KBM)
40  ZSN=Z(1,KBM)
40  FX=FX+.5*(STRESS(1)+STRESS(1))*CON
40  AMY=AMY+(STRESS(1)*(ZSN+2.*ZS)+STRESS(1)*(ZS+2.*ZSN))*CON/6.
40  AMZ=AMZ+(STRESS(1)*(YSN+2.*YS)+STRESS(1)*(YS+2.*YSN))*CON/6.
30  CONTINUE
40  AMZ=AMZ
40  AMX=TORCON(KBM)*E(2)*ROT/(2.*(1.+E(7)))
40  RETURN
40  END

```

40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.

```

1. SUBROUTINE VECTOR(XC,YC,ZC,UD,NODE,JE,DIST,S,IP,A,B,C,ABC)
2. DIMENSION XC(1),YC(1),ZC(1),UD(1),NODE(9,1),IP(1),DIST(1)
3. DIMENSION X(3),Y(3),Z(3)
4. DO 105 M=1,3
5. N=NODE(M,JE)
6. J=6*(N-1)
7. X(M)=XC(N)+UD(J+1)
8. Y(M)=YC(N)+UD(J+2)
9. Z(M)=ZC(N)+UD(J+3)
10. S=0.
11. DO 110 M=1,3
12. N=IP(M+1)
13. DIST(M)=SQRT((X(M)-X(N))**2+(Y(M)-Y(N))**2+(Z(M)-Z(N))**2)
14. S=S+DIST(M)
15. S=0.5*S
16. A=Y(1)*(Z(2)-Z(3))+Y(2)*(Z(3)-Z(1))+Y(3)*(Z(1)-Z(2))
17. B=X(1)*(Z(3)-Z(2))+X(2)*(Z(1)-Z(3))+X(3)*(Z(2)-Z(1))
18. C=X(1)*(Y(2)-Y(3))+X(2)*(Y(3)-Y(1))+X(3)*(Y(1)-Y(2))
19. ABC=SQRT(A**2+B**2+C**2)
20. RETURN
21. ENTRY VECT(THETA,XCOMP,YCOMP,ZCOMP)
22. XCOMP=((X(2)-X(1))*COS(THETA)+(B*(Z(2)-Z(1))-C*(Y(2)-Y(1)))
23. **SIN(THETA))/ABC)/DIST(1)
24. YCOMP=((Y(2)-Y(1))*COS(THETA)+(C*(X(2)-X(1))-A*(Z(2)-Z(1)))
25. **SIN(THETA))/ABC)/DIST(1)
26. ZCOMP=((Z(2)-Z(1))*COS(THETA)+(A*(Y(2)-Y(1))-B*(X(2)-X(1)))
27. **SIN(THETA))/ABC)/DIST(1)
28. RETURN
29. END

```



```

1. SUBROUTINE READIN(NELE,NNODE,NDGREE,MUD,NUMDIS,NODDIS,
2. *E,XC,YC,ZC,NODE,ANGLE,UD)
3. COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6),
4. *STRS(10000)
5. COMMON/DYNAM/DELT,TIMEND,MXSTEP,NTSTEP,TIME,EYMED
6. COMMON/OUTPA/NPRU,NPRS,NPFREQ,NPIC,OUT(9,40),NPOUT(2,10),
7. 1 TITLE(20),NPRV,NPRP
8. COMMON/CONTRL/KONTRL(10)
9. COMMON /BEAMS/RMASS(3,3,100),DICOS(3,3,150),BAR(1200),
10. *PMASS(4,100),NSECT
11. DIMENSION XC(1),YC(1),ZC(1),E(16,1),NODE(9,1),NODDIS(1),UD(1),
12. *JJ(4),ANGLE(3,1)
13. DIMENSION IHD(10),HD(10)
14. EQUIVALENCE (INDEX(1),IHD(1)),(INDEX(11),HD(1))
15. INTEGER OUT
16. READ(5,1000,END=19191) TITLE
17. 1000 FORMAT(20A4)
18. WRITE(6,2000) TITLE
19. 2000 FORMAT(1H1/10X,20A4)
20. READ (5,1011) NNODE,NELE,NUMMAT,NUMDIS,MXSTEP,NDGREE,DELT,NCORD,
21. *NSECT
22. 1011 FORMAT (6I5,E10.6,2I5)
23. WRITE (6,2001) NNODE,NELE,NUMMAT,NUMDIS,MXSTEP,NDGREE,DELT,NCORD,
24. *NSECT
25. 2001 FORMAT(1H0,
26. 1 4X,'NUMBER OF NODES',I10/,
27. 2 5X,'NUMBER OF ELEMENTS',I10/,
28. 3 5X,'NUMBER OF MATERIALS',I10/,
29. 4 5X,'NO. OF DISP. NODES',I10/,
30. 5 5X,'MAX.NO. OF TIME STEPS',I10/,
31. 6 5X,'NO. OF DIMENSIONS',I10/,
32. 7 5X,'TIME INCREMENT',I1PE16.5/,
33. 8 5X,'NO. OF COORD NODES',I10/,
34. 9 5X,'NO. OF BEAM XSECT',I10/)
35. READ(5,3010)(KONTRL(I),I=1,10)
36. 3010 FORMAT(10I5)
37. IF(KONTRL(7).EQ.0)KONTRL(7)=2
38. I1=KONTRL(7)
39. WRITE(6,3011)(KONTRL(1),I=1,10)

```

```

40. 3011 FORMAT(1H0'KONTRL(I)='1,1015)
41. C
42. C
43. C
44. C
45. C
46. C
47. C
48. C
49. C
50. C
51. C
52. C
53. C
54. C
55. C
56. C
57. C
58. C
59. C
60. C
61. C
62. C
63. C
64. C
65. C
66. C
67. C
68. C
69. C
70. C
71. C
72. C
73. C
74. C
75. C
76. C
77. C
78. C

      INTERPRETATION OF KONTRL VECTOR

      KONTRL(3)      EXPANSION FACTOR IN MESH GENERATOR
      KONTRL(5)      BETAX1000 INTEGRATION PARAMETER =0
      KONTRL(6)      ITERATION COUNTER =0
      KONTRL(7)      NUMBER OF CROSS-SECTION INTEGRATION POINTS
                      =2 IF ZERO (MINIMUM VALUE)
      KONTRL(8)      CHECKPOINT INTERVAL =0 NO OPERATION
                      \0 MAKE CHECKPOINT
                      )0 RESTART AND CHECKPOINT

      KONTRL(9)      RESTART CHECKPOINT NUMBER
      KONTRL(10)     PLOT FLAG = MAKE OUTPUT FILE IF NE 0

      10 UNIT DESIGNATIONS

      LOGICAL UNIT 20  MESH PLOT DATA
      LOGICAL UNIT 25  CHECKPOINT/RESTART DATA
      LOGICAL UNIT 22  TIME HISTORY DATA FOR POSTPROCESSING

      DO 50 I=1,NUMMAT
      READ(5,1050) MTYPE
      READ(5,1051) (E(J,MTYPE),J=1,16)
      WRITE(6,2050) MTYPE,(E(J,MTYPE),J=1,16)

      50 CONTINUE
      1050 FORMAT(15,E10.4)
      1051 FORMAT(8E10.4)
      2050 FORMAT(1H0,4X,'MATI',I3,10X/3X,'E=MATRIX',1P8E15.5/
      1 11X,1P8E15.5)

      GET BEAM CROSS SECTION DATA

```

```

79. C
80. C
81. C
82. C
83. C
84. C
85. C
86. C
87. C
88. C
89. C
90. C
91. C
92. C
93. C
94. C
95. C
96. C
97. C
98. C
99. C
100. C
101. C
102. C
103. C
104. C
105. C
106. C
107. C
108. C
109. C
110. C
111. C
112. C
113. C
114. C
115. C
116. C
117. C

IF(NSECT.GT.0) CALL BGEOM(NSECT)
READ AND PRINT NODAL POINT DATA
59 WRITE(6,2003)
2003 FORMAT(1H0//8X,'NODE NO.',12X,'X=ORDINATE',3X,'Y=ORDINATE',3X,
1'Z=ORDINATE')
L=0
NTOT=NNODE+NCORD
60 READ(5,1003)N,XC(N),YC(N),ZC(N),(PMASS(I,N),I=1,4)
1003 FORMAT(15,5X,7E10.4)
NL=L+1
IF(L.EQ.0) GO TO 70
NML=N-L
DS=NML
IF(KONTRL(3).EQ.0) GO TO 69
FAC=KONTRL(3)
FAC=FAC/100.0
DS=0.0
DO 641 I=1,NML
641 DS=DS+FAC*(I-1)
69 DXC=(XC(N)-XC(L))/DS
DYC=(YC(N)-YC(L))/DS
DZC=(ZC(N)-ZC(L))/DS
70 L=L+1
IF(N-L)100,90,80
80 XC(L)=XC(L-1)+DXC
YC(L)=YC(L-1)+DYC
ZC(L)=ZC(L-1)+DZC
IF(KONTRL(3).EQ.0) GO TO 70
DXC=DXC*FAC
DYC=DYC*FAC
DZC=DZC*FAC
GO TO 70
90 WRITE(6,2004) (K,XC(K),YC(K),ZC(K),K=NL,N)
IF(NTOT=N)100,110,60
100 WRITE(6,2005)N
CALL EXIT
110 CONTINUE

```

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700

```



```

157. 195  ANGLE(J,I)=0.0174533*ANGLE(J,I)
158. 2202  FORMAT(5X,I5,5X,I4,4X,3I2,3X,3F10.3)
159. 200  CONTINUE
160. C
161. READ OUTPUT CODE CARDS
162. WRITE (6,1404)
163. 1404  FORMAT (1H0//5X,'OUTPUT CODE')
164. READ (5,403) NPFREQ,NPRU,NPRS,NPRV,NPRP,NPIC
165. 403  FORMAT (6I10)
166. WRITE(6,1405) NPFREQ,NPRU,NPRS,NPRV,NPRP,NPIC
167. 1405  FORMAT (1H0,9X,'NPFREQ=',I10/
168. 1 10X,'NPRU =',I10/
169. 2 10X,'NPRS =',I10/
170. 3 10X,'NPRV =',I10/
171. 4 10X,'NPRP =',I10/
172. 5 10X,'NPIC =',I10)
173. IF (NPRU.EQ. 0) GO TO 411
174. READ (5,400) ((OUT(J,I),J=1,9),I=1,NPRU)
175. 400  FORMAT(I10,5A4,3E10.0)
176. WRITE (6,401) (I,(OUT(J,I),J=1,9),I=1,NPRU)
177. 401  FORMAT (1H0,5X,'DISPLACEMENT AND VELOCITY OUTPUT',
178. 1(4X,I3,I10,5X,5A4,3E15.4))
179. DO 410 I=1,NPRU
180. CALL DECOD (OUT(1,I),JJ)
181. IF (JJ(4) .GT. NNODE) GO TO 405
182. 410 CONTINUE
183. 411 IF (NPRS.EQ. 0) GO TO 215
184. I3=NPRU+1
185. I4=NPRU+NPRS
186. READ (5,400) ((OUT(J,I),J=1,9),I=I3,I4)
187. WRITE (6,402) (I,(OUT(J,I),J=1,9),I=I3,I4)
188. 402  FORMAT(1H0,5X,'STRESS OUTPUT',(4X,I3,I10,5X,5A4,3E15.4))
189. DO 521 I=I3,I4
190. CALL DECOD (OUT(1,I),JJ)
191. 521 CONTINUE
192. 215 CONTINUE
193. 298 IF (NPIC .LE. 0) GO TO 512
194. WRITE(6,589)
195. 589  FORMAT(1H0,5X,'INPUT PICTURE OUTPUT',12X,'AT STEP',10X,'CODE')
      DO 591 K=1,NPIC

```



```

196.      READ(5,581) (NPOUT(I,K),I=1,2)
197.      FORMAT(2I10)
198.      WRITE(6,582) (NPOUT(I,K),I=1,2)
199.      FORMAT(10X,I7,10X,I4)
200.      J=0
201.      C
202.      C
203.      C
204.      C
205.      C
206.      C
207.      C
208.      C
209.      C
210.      C
211.      C
212.      C
213.      C
214.      C
215.      C
216.      C
217.      C
218.      C
219.      C
220.      C
221.      C
222.      C
223.      C
224.      C
225.      C
226.      C
227.      C
228.      C
229.      C
230.      C
231.      C
232.      C
233.      C
234.      C

      OUTPUT FOR MESH AND DEFORMED STRUCTURE PLOTS
      CHECK RESTART FLAG, SKIP AND POSITION FILE IF NECESSARY

      REWIND 20
      IF(KONTRL(9).NE.0) GO TO 850
      WRITE(20,1060)(K,XC(K),YC(K),ZC(K),K=1,NNODE)
      DO 856 K=1,NELE
      DO 857 I=1,4
      IHD(I)=NODE(I,K)
      IF(NODE(8,K).LE.1) GO TO 858
      IHD(3)=0
      IHD(4)=0
      CONTINUE
      WRITE(20,1070) K,(IHD(I),I=1,4)
      CONTINUE
      1060 FORMAT(I5,3E12.5)
      1070 FORMAT(5I5)
      GO TO 851
      850  IF(NPIC.LE.0) GO TO 851
      KREC=KONTRL(9)*IABS(KONTRL(8))
      J=0
      DO 853 I=1,NPIC
      IF(KREC.GE.NPOUT(1,I))J=I
      CONTINUE
      853  DO 852 K=1,NNODE
      852  READ(20,1060)KREC,(HD(I),I=1,3)
      DO 854 K=1,NELE
      854  READ(20,1070)(IHD(I),I=1,5)
      IF(J.EQ.0) GO TO 851

```

235.	J=J*NNODE	00023500
236.	DO 855 K=1,J	00023600
237.	855 READ(20,1060)KREC,(HD(I),I=1,3)	00023700
238.	851 CONTINUE	00023800
239.		00023900
240.		00024000
241.	INITIALIZE TIME HISTORY PLOT FILE IF REQUIRED	00024100
242.		00024200
243.	IF(KONTRL(10).EQ.0) GO TO 800	00024300
244.	REWIND 22	00024400
245.		00024500
246.	CHECK RESTART FLAG AND POSITION FILE IF NECESSARY	00024600
247.		00024700
248.	IF(KONTRL(9).EQ.0) GO TO 801	00024800
249.		00024900
250.	POSITION FILE - THREE HEADER RECORDS	00025000
251.		00025100
252.	DO 802 I=1,3	00025200
253.	802 READ(22) RECORD	00025300
254.	J=KONTRL(9)*IABS(KONTRL(8))	00025400
255.	DO 803 I=1,J	00025500
256.	803 READ(22) RECORD	00025600
257.	GO TO 800	00025700
258.	801 CONTINUE	00025800
259.		00025900
260.	WRITE HEADER RECORDS	00026000
261.		00026100
262.	NTOT=NP RU+NP RS	00026200
263.	IHD(1)=NTOT	00026300
264.	IHD(2)=NP RU	00026400
265.	IHD(3)=NP RS	00026500
266.	IHD(4)=MXSTEP	00026600
267.	IHD(5)=MXSTEP	00026700
268.	IHD(6)=0	00026800
269.	IHD(7)=0	00026900
270.	IHD(8)=NELE	00027000
271.	HD(1)=DELT	00027100
272.	WRITE(22) IHD,HD,TITLE	00027200
273.	WRITE(22)(NODE(5,I),I=1,NELE)	00027300

274.
275.
276.
277.
278.
279.
280.
281.
282.
283.

```
      WRITE(22)(OUT(1,I),I=1,NTOT)
      800 CONTINUE
C
      IF ( NPFREQ .LE. 0 )  NPFREQ=1
      RETURN
      405 WRITE (6,2410 )
      2410 FORMAT (1H0,5X,'ERROR IN OUTPUT DATA')
      19191 CALL EXIT
      RETURN
      END
```

00027400
00027500
00027600
00027700
00027800
00027900
00028000
00028100
00028200
00028300

```

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.

SUBROUTINE FREEFD(KONWAY,NUMDIS,NODDIS,NNODE,NDGREE,XC,YC,ZC,UD,
*UD1,UD2,FORCD,E,IX,IP)
  DIMENSION ZC(1),E(16,1),IX(9,1),IP(1),UD2(1)
  DIMENSION UD(1),UD1(1),FORCD(1),NODDIS(1),XC(1),YC(1)

  LOADING FOR 4 NODE CANTILEVER BEAM PROBLEM--6 DOF
    10 LB ON NODE FOUR IN Z-DIRECTION

  FORCD(21)=10.
  RETURN
  END

C
C
C
C
C

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200

```

```

1. SUBROUTINE OUTPUT
2. COMMON/WORK/ANGLE(600)
3. COMMON XC(300),YC(300),ZC(300),UD(900),UD1(900),UD2(900),
4. *FORCD(900),E(16,5),
5. *IX(9,300),NODDIS(100),NUMNP,NUMEL,NDGREE,MUD,NUMDIS
6. COMMON/DYNAM/DELTA,TIMEND,MXSTEP,NTSTEP,TIME,EYMED
7. COMMON/OUTPA/NPRU,NPRS,NPFREQ,NPIC,OUT(9,40),NPOUT(2,10),
8. 1 TITLE(20),NPRV,NPRP
9. COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6),
10. *STRS(10000)
11. COMMON/CONTRL/KONTRL(10)
12. DIMENSION UU(20),SS(20),PSU(1,1),AUX(900),JJ(4),D(3)
13. EQUIVALENCE (NUMNP,NNODE),(NUMNP,NUMHAL),(NUMEL,NELE)
14. INTEGER OUT
15. IF ( NPRU .LE. 0 .AND. NPRS .LE. 0 .AND. NPIC .LE. 0 ) GOTO 405
16. NP=MOD(NTSTEP,NPFREQ)
17. KKK=0
18. NKN=NTSTEP/NPFREQ
19. IF(NP.EQ.0)WRITE(6,103)NTSTEP,TIME
20. 103 FORMAT(1X,'CYCLE=',I5,4X,'TIME=',1PE12.6)
21. IF ( NPRU .LE. 0 ) GO TO 104
22. DO 151 I=1, NPRU
23. CALL DECOD ( OUT(1,I),JJ)
24. II=(JJ(4)-1)*NDGREE +JJ(1)
25. IF ( JJ(2) .EQ. 0 ) GO TO 1
26. IF ( JJ(2) .EQ. 1 ) GO TO 2
27. IF ( JJ(2) .EQ. 2 ) GO TO 3
28. 1 UU(I)=UD(II)
29. GO TO 15
30. 2 UU(I)=UD1(II)
31. GO TO 15
32. 3 UU(I)=UD2(II)
33. 15 CONTINUE
34. IF ( JJ(3) .EQ. 0 ) GO TO 151
35. KKK=KKK+1
36. PSU(NKN,KKK)=UU(I)
37. CONTINUE
38. 151 IF(NP.EQ.0) WRITE(6,102)(UU(J),J=1,NPRU)
39. DO 550 J=1,NPRU

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002800
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900

```


00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800

```

550  AUX(J)=UU(J)
102  FORMAT ( 25X,1P6E16.6 )
104  CONTINUE
      IF ( NPRS .LE. 0 ) GO TO 220
      I3=NPRU+1
      I4=NPRU+NPRS
      I=1
      NPRS1=NPRS
      DO 221 II=I3,I4
      CALL DECOD ( OUT(1,II),JJ)
      J4=JJ(4)/10
      IF(NDGREE.EQ.2) GO TO 222
      J5=JJ(4)-10*J4
      K2=JJ(4)/10000
      IF(K2.EQ.0) GO TO 105
      J3=JJ(4)-10000*K2
      J4=J3/10
      J5=J3-10*J4
      K5=J4/100
      K3=K2/10
      K1=K2-10*K3
      K4=K3+10*K5
      IND=INDEX(K4)-1
      M1=IND+27*(J5-1)+3*(K1-1)+1
      M2=M1+9
      GX=SQRT((STRS(M1)-STRS(M1+1))*2+STRS(M1+2)**2)
      TX=0.5*SQRT((STRS(M2)-STRS(M2+1))*2+4.*STRS(M2+2)**2)
      IF(JJ(2).LT.8) GO TO 105
      CALL VECTOR(XC,YC,ZC,UD,IX,K4,D,S,IP,A,B,C,ABC)
      THETA=ACOS(SQRT(S*(S-D(2))/(D(1)*D(3))))
      ST=SIN(THETA)
      CT=COS(THETA)
105  IND=INDEX(J4)-1
      N1=IND+27*(J5-1)+3*(JJ(1)-1)+1
      N2=N1+9
      GAMMAX=SQRT((STRS(N1)-STRS(N1+1))*2+STRS(N1+2)**2)
      TAUMAX=0.5*SQRT((STRS(N2)-STRS(N2+1))*2+4.*STRS(N2+2)**2)
      JJ=JJ(2)+1
      GO TO(110,111,112,113,114,115,116,117,118,119), JJ

```

40.
41.
42.
43.
44.
45.
46.
47.
48.
49.
50.
51.
52.
53.
54.
55.
56.
57.
58.
59.
60.
61.
62.
63.
64.
65.
66.
67.
68.
69.
70.
71.
72.
73.
74.
75.
76.
77.
78.

```

79. 110 NPRS1=NPRS1+2
80. CALL VECTOR(XC,YC,ZC,UD,IX,J4,D,S,IP,A,B,C,ABC)
81. BOTTOM=STRS(N1)+STRS(N1+1)
82. THETA=.78539816
83. IF(BOTTOM.EQ.0.) GO TO 108
84. THETA=STRS(N1+2)/BOTTOM
85. THETA=ACOS(SQRT(S*(S-D(2))/(D(1)*D(3))))+0.5*ATAN(THETA)
86. CALL VECT(THETA,XCOMP,YCOMP,ZCOMP)
87. SS(I)=XCOMP
88. SS(I+1)=YCOMP
89. SS(I+2)=ZCOMP
90. I=I+2
91. GO TO 223
92. 111 SS(I)=0.5*(STRS(N1)+STRS(N1+1))+0.5*GAMMAX
93. IF(K2.NE.0) SS(I)=0.5*(SS(I)+0.5*(STRS(M1)+STRS(M1+1)+GX))
94. GO TO 223
95. 112 SS(I)=0.5*(STRS(N1)+STRS(N1+1))+0.5*GAMMAX
96. IF(K2.NE.0) SS(I)=0.5*(SS(I)+0.5*(STRS(M1)+STRS(M1+1)+GX))
97. GO TO 223
98. 113 SS(I)=GAMMAX
99. IF(K2.NE.0) SS(I)=0.5*(SS(I)+GX)
100. GO TO 223
101. 114 SS(I)=0.5*(STRS(N2)+STRS(N2+1))+TAUMAX
102. IF(K2.NE.0) SS(I)=0.5*(SS(I)+0.5*(STRS(M2)+STRS(M2+1))+TX)
103. GO TO 223
104. 115 SS(I)=0.5*(STRS(N2)+STRS(N2+1))-TAUMAX
105. IF(K2.NE.0) SS(I)=0.5*(SS(I)+0.5*(STRS(M2)+STRS(M2+1))-TX)
106. GO TO 223
107. 116 SS(I)=TAUMAX
108. IF(K2.NE.0) SS(I)=0.5*(SS(I)+TX)
109. GO TO 223
110. 117 SS(I)=STRS(N1+18)
111. GO TO 223
112. 118 CALL VECTOR(XC,YC,ZC,UD,IX,J4,D,S,IP,A,B,C,ABC)
113. THETA=ACOS(SQRT(S*(S-D(2))/(D(1)*D(3))))
114. 1+STRS(N1+2)*SIN(THETA)*COS(THETA)
115. IF(K2.NE.0) SS(I)=0.5*(SS(I)+STRS(M1)*CT**2+STRS(M1+1)*ST**2)
116. 1+STRS(M1+2)*ST*CT)
117. GO TO 223
00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700

```

```

118. 119 CALL VECTOR(XC,YC,ZC,UD,IX,J4,D,S,IP,A,B,C,ABC)
119. THETA=1.5707963=ACOS(SQRT(S*(S=D(2))/(D(1)*D(3))))
120. 1+STRS(N1+2)*SIN(THETA)*COS(THETA)
121. IF(K2.NE.0) SS(I)=0.5*(SS(I)+STRS(M1)*CT**2+STRS(M1+1)*ST**2
122. 1+STRS(M1+2)*ST*CT)
123. GO TO 223
124. 222 J12=10*JJ(1)+JJ(2)
125. IND=INDEX(J4)=1
126. SS(I)=STRS(IND+J12)
127. 223 IF(JJ(3).EQ.0) GO TO 221
128. KKK=KKK+1
129. PSU(NKN,KKK)=SS(I)
130. 221 I=I+1
131. IF(NP.EQ.0) WRITE(6,102)(SS(J),J=1,NPRS1)
132. DO 551 J=1,NPRS
133. NXT=NPRU+J
134. AUX(NXT)=SS(J)
135. 220 CONTINUE
136. IF(KONTRL(10).EQ.0) GO TO 552
137.
138.
139. WRITE POST PROCESSOR FILE
140.
141. NTOT=NPRU+NPRS
142. WRITE(22)(AUX(J),J=1,NTOT)
143. CONTINUE
144. IF ( NPIC .LE. 0 ) GO TO 109
145. DO 129 I=1,NPIC
146. IF ( NTSTEP .NE. NPOUT(1,I) ) GO TO 129
147. WRITE(6,302) TIME
148. 302 FORMAT('1', 9X, 'TIME=',E20.6/
149. 1 10X, 'DISPLACEMENTS AT NODE',10X, 'IN X-DIRECTION',10X,
150. 2 'IN Y-DIRECTION',10X, 'IN Z-DIRECTION')
151. N=NDGREE
152. DO 130 LD=1,NUMHAL
153. L=NDGREE*(LD-1)
154. 130 WRITE(6,301) LD,(UD(L+J),J=1,NDGREE)
155. 301 FORMAT(10X,I5,5X,6E12.5)
156.

```

```

157.      C      OUTPUT FOR DEFKRMED MESH PLOTS
158.      C
159.      DO 1010 J=1,NUMHAL
160.      L=(J-1)*NDGREE
161.      WRITE(20,1020)J,UD(L+1),UD(L+2),UD(L+3)
162.      1010  FORMAT(I5,3E12,5)
163.      129  CONTINUE
164.      109  CONTINUE
165.      IF ( NTSTEP.LT. MXSTEP ) RETURN
166.      IF ( KKK.EQ. 0 ) RETURN
167.      405  WRITE(6,2410)
168.      2410 FORMAT(1H0,5X,'ERROR IN OUTPUT DATA')
169.      CALL EXIT
170.      RETURN
171.      END
00015700
00015800
00015900
00016000
00016100
00016200
00016300
00016400
00016500
00016600
00016700
00016800
00016900
00017000
00017100

```

```

1. SUBROUTINE RESTRT(IFLAG,NREC,ICLK)
2. COMMON/JUNK/TL(9,300)
3. COMMON/DYNAM/DELT,TIMEND,MXSTEP,NTSTEP,TIME
4. COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6),
5. *STRS(10000)
6. COMMON XC(300),YC(300),ZC(300),UD(900),UD1(900),UD2(900)
7. IF(IFLAG)10,20,30
8. 10 REWIND 25
9. IF(NREC.EQ.0)RETURN
10. DO 40 I=1,NREC
11. 40 READ(25)NTSTEP,TIME,UD,UD1,UD2,STRS,TL,ICLK
12. WRITE(6,60)NTSTEP
13. 60 FORMAT(/5X,'*** RESTART AT TIME STEP ',I4,' ***'//)
14. RETURN
15. 30 ICLK=ICLK+1
16. WRITE(6,50)ICLK
17. 50 FORMAT(/5X,'*** CHECKPOINT NO. ',I4,' ***'//)
18. WRITE(25)NTSTEP,TIME,UD,UD1,UD2,STRS,TL,ICLK
19. 20 RETURN
20. END

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000

```



```

1. SUBROUTINE ASSBLE(NUMEL,NUMNP,NDGREE,IX,XC,YC,ZC,E,SMASS,STRS,
2. *STRESS,STRAIN,INDEX,I1,8SIDE,IP)
3. COMMON/JUNK/TL(9,300),POINT(3,3,300)
4. COMMON /BEAMS/RMASS(3,3,100),DICOS(3,3,150),BAK(1200),
5. *PMASS(4,100),NSECT
6. DIMENSION EULCO(3,3,300)
7. DIMENSION TEMP(9)
8. EQUIVALENCE (TL(1,1),EULCO(1,1,1))
9. DIMENSION STRS(1),STRESS(1),8TRAIN(1),SIDE(1),IP(1)
10. DIMENSION IX(9,1),XC(1),YC(1),ZC(1),E(16,1),SMASS(1),INDEX(1)
11. MEG=NUMNP*NDGREE
12. DO 100 JE=1,MEG
13. 100 SMASS(JE)=0.0
14. DO 101 I=1,NUMNP
15. DO 101 J=1,3
16. DO 101 K=1,3
17. 101 RMASS(J,K,I)=0.
18. INDEX(1)=1
19. I3=I*3
20. DO 105 I=1,3
21. DO 104 J=1,3
22. K=15*(I-1)+J
23. STRAIN(K)=0.0
24. 104 STRESS(K)=0.0
25. IP(I)=I
26. 105 IP(I+3)=I
27. WRITE(6,1)
28. 1 FORMAT(//5X,'PERIOD OF VIBRATION FOR BEAMS',//4X,'ELEMENT',3X,
29. 1'Elongation',3X,'Torsion',3X,'ZZ-MOM ',3X,'ZZ-SHR ',
30. 23X,'YY-MOM ',3X,'YY-SHR ',//)
31. DO 115 JE=1,NUMEL
32. MTYPE=IX(9,JE)
33. KEL=IX(08,JE)
34. IND=INDEX(JE)
35. IF(KEL.GT.1) GO TO 200
36. CALL TASME(JE,NDGREE,IX,XC,YC,ZC,E,SMASS,SIDE(3*JE=2),IP,
37. *STRS(IND),NTOT,I1,DICOS(1,1,JE),RMASS)
38. GO TO 210
39. 200 CONTINUE

```

```

00000100
00000200
00000300
00000400
00000500
00000600
00000700
00000800
00000900
00001000
00001100
00001200
00001300
00001400
00001500
00001600
00001700
00001800
00001900
00002000
00002100
00002200
00002300
00002400
00002500
00002600
00002700
00002900
00003000
00003100
00003200
00003300
00003400
00003500
00003600
00003700
00003800
00003900

```

```

40. CALL BASME(JE,NDGREE,IX,XC,YC,ZC,E,SMASS,DICUS,RMASS,PMASS,
41. *I1,STRS(IND),NTOT)
42. 210 CONTINUE
43. INDEX(JE+1)=INDEX(JE)+NTOT
44. 115 CONTINUE
45. C
46. C INITIALIZE THE LAMBDA TRANSFORMATION MATRIX
47. C
48. DO 150 I=1,NUMNP
49. DO 150 J=1,9
50. 150 TL(J,I)=0.0
51. DO 125 I=1,NUMNP
52. TL(1,I)=1.0
53. TL(5,I)=1.0
54. 125 TL(9,I)=1.0
55. DO 250 I=1,NUMNP
56. DO 260 J=1,3
57. 260 RMASS(J,J,I)=RMASS(J,J,I)+PMASS(J+1,I)
58. TEMP(1)=RMASS(1,1,I)
59. TEMP(2)=RMASS(1,2,I)
60. TEMP(3)=RMASS(2,2,I)
61. TEMP(4)=RMASS(3,1,I)
62. TEMP(5)=RMASS(3,2,I)
63. TEMP(6)=RMASS(3,3,I)
64. CALL EIGEN(TEMP,TL(1,I),TL(4,I),TL(7,I),DUMA,DUMB,0)
65. CALL CROSS(TL(1,I),TL(4,I),TL(7,I),DUMA,DUMB,0)
66. DO 253 J=1,3
67. DO 253 K=1,3
68. 253 RMASS(J,K,I)=0.
69. RMASS(1,1,I)=TEMP(1)
70. RMASS(2,2,I)=TEMP(3)
71. RMASS(3,3,I)=TEMP(6)
72. 250 CONTINUE
73. DO 220 I=1,NUMNP
74. NN=6*(I-1)
75. DO 230 J=1,3
76. 230 SMASS(NN+J)=SMASS(NN+J)+PMASS(1,I)
77. NN=NN+3
78. DO 220 J=1,3

```

```

00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800

```

```

79. 220 SMASS(NN+J)=RMASS(J,J,I)
80.
81. C
82. C
83. C
84. C
85. C
86.
87.
88.
89.
90.
91.
92.
93.
94.
95.
96.
97.
98.
99.
100.
101.
102.
103.
104.
105.
106.
107.
108.
109.
110.
111.
112.
113.
114.
115.
116.
117.

      CONSTRUCT INITIAL TL(TRANSP)*MU(HAT)

      MJ = 0
      DO 50 I = 1,NUMEL
        KEL=IX(8,I)
        IF(KEL.GT.1) GO TO 300
        DO 310 JJ=1,3
          NN=IX(JJ,I)
          CALL GTPRD(TL(1,NN),DICOS(1,3,I),POINT(1,JJ,NN),3,3,1)
310      CONTINUE
300      GO TO 50
300      CONTINUE
      DO 51 J1 = 1,2
        K = IX(J1,I)
        KK=IX(J1+2,I)
        DO 52 L = 1,2
          DO 53 M = 1,3
            SUM = 0.0
            MJ = MJ+ 1
            DO 54 N = 1,3
              S4 SUM = SUM + EULCO(N,M,KK)*DICOS(N,L,I)
              S3 BAR(MJ) = SUM
              S2 CONTINUE
              BAR(MJ+1)=XC(K) - XC(KK)
              BAR(MJ+2)=YC(K) - YC(KK)
              BAR(MJ+3)=ZC(K) - ZC(KK)
              MK=MJ
              MJ=MJ+3
              DO 55 NN=1,3
                MJ=MJ+1
                S5 BAR(MJ)=EULCO(1,NN,KK)*BAR(MK+1) + EULCO(2,NN,KK)*BAR(MK+2)
                  + EULCO(3,NN,KK)*BAR(MK+3)
                S1 CONTINUE
                S5 CONTINUE
                S5 LSTRS=INDEX(NUMEL+1)-1

```

118.
119.
120.
121.
122.

WRITE(6,400) LSTRS,MJ
400 FORMAT(///10X,' STRESS AND BAR STORAGE
*15///)
RETURN
END

LSTRS',I5,' LBAR',

00011800
00
00012000
00012100
00012200

```

1. SUBROUTINE SOLVE(NUMEL,NUMNP,NDGREE,NUMDIS,NODDIS,SMASS,XC,YC,ZC, 00000100
2. *IX,E,XO,X1,V,AO,A1,FORCD,FINT,ANGLE) 00000200
3. DYNAMIC TRANSIENT SOLUTION WITH NEWMARK BETA METHOD 00000300
4. COMMON/JUNK/TL(9,300) 00000400
5. COMMON/CONTRL/KONTRL(10) 00000500
6. COMMON/STR/I1,INDEX(300),STRAIN(45),STRESS(45),SIDE(900),IP(6), 00000600
7. *STRS(10000) 00000700
8. COMMON/DYNAM/DELT,TIMEND,MXSTEP,NTSTEP,TIME,EYMED 00000800
9. DIMENSION FINT(1) 00000900
10. DIMENSION OLDS(600) 00001000
11. DIMENSION ANGLE(3,1) 00001100
12. DIMENSION XO(1),X1(1),V(1),AO(1),A1(1),FORCD(1),NODDIS(1),SMASS(1) 00001200
13. *,XC(1),YC(1),IX(9,1),E(16,1),JJ(4),ZC(1) 00001300
14. DATA EPSI/1.E=3/ 00001400
15. DO 956 I=1,3 00001500
16. IP(I)=I 00001600
17. IP(I+3)=I 00001700
18. I2=I1*2 00001800
19. I4=I1*4 00001900
20. I6=I1*6 00002000
21. ICHK=0 00002100
22. CONTROL VARIABLES FOR INTEGRATION 00002200
23. BETA=KONTRL(5) 00002300
24. BETA=BETA/1000. 00002400
25. ITERMX=KONTRL(6) 00002500
26. MEQ=NUMNP*NDGREE 00002600
27. IF (BETA.GT.0.5) JSTOP=1 00002700
28. IF (ITERMX.GT.5) JSTOP=1 00002800
29. IF (JSTOP.EQ.1) GO TO 980 00002900
30. ZERO DEPENDENT VARIABLES 00003000
31. DO 212 I=1,MEQ 00003100
32. FORCD(I)=0.0 00003200
33. AO(I)=0.0 00003300
34. A1(I)=0.0 00003400
35. XO(I)=0. 00003500
36. X1(I)=0. 00003600
37. 212 V(I)=0. 00003700
38. DO 732 I=1, I2 00003800
39. STRAIN(I)=0.0 00003900

```



```

40. 732 STRESS(I)=0.0
41. DO 733 I=1,600
42. 733 OLDS(I)=0.
43. COMPUTE INTEGRATION PARAMETERS
44. TIME=0.0
45. NTSTEP=0
46. C1=(0.5-BETA)*DELT*DELT
47. C2=BETA*DELT*DELT
48. C3=DELT/2.0
49. C INITIAL CONDITIONS
50. CALL FREEFD(2,NUMDIS,NODDIS,NUMNP,NDGREE,
51. *XC,YC,ZC,XO,V,AO,FORCD,E,IX,IP)
52. CALL FRCIN(NUMEL,NUMNP,NDGREE,IX,E,XO,FINT,OLDS)
53. C INTEGRATION LOOP
54. IF(KONTRL(8).GE.0)GO TO 200
55. CALL RESTRT(-1,KONTRL(9),ICLK)
56. KONTRL(8)=-KONTRL(8)
57. 200 CONTINUE
58. 400 ITER=0
59. NTSTEP=NTSTEP+1
60. TIME=TIME+DELT
61. 401 DO 410 I=1,MEQ
62. FORCD(I)=0.0
63. XO(I)=X1(I)
64. 410 A1(I)=AO(I)
65. CALL FREEFD(1,NUMDIS,NODDIS,NUMNP,NDGREE,
66. *XC,YC,ZC,XO,V,AO,FORCD,E,IX,IP)
67. 415 ITER=ITER+1
68. C ALTER PRESCRIBED DISPLACEMENTS
69. C ZERO ALL ACC. AND VEL. CORRESPONDING TO PRESCRIBED DISPLACEMENT
70. DO 430 I=1, NUMDIS
71. CALL DEQOD(NODDIS(I),JJ)
72. J4=JJ(4)
73. MLREF=NDGREE*(J4-1)
74. IF(JJ(1).LE.2) GO TO 510
75. MLREF=MLREF+3
76. DO 500 J=1,3
77. JJ(J)=JJ(J)-3
78. 500 JJ(J)=JJ(J)-3
510 CONTINUE
00004000
00004100
00004200
00004300
00004400
00004500
00004600
00004700
00004800
00004900
00005000
00005100
00005200
00005300
00005400
00005500
00005600
00005700
00005800
00005900
00006000
00006100
00006200
00006300
00006400
00006500
00006600
00006700
00006800
00006900
00007000
00007100
00007200
00007300
00007400
00007500
00007600
00007700
00007800

```

```

79. DO 430 K=1,3
80. IF(ANGLE(K,I).EQ.0.)GO TO 4202
81. CALL ROTATE(ANGLE(K,I),J4,XO,NDGREE,1)
82. CALL ROTAT ( X1,1 )
83. CALL ROTAT ( V,1 )
84. CALL ROTAT ( A1,1 )
85. CALL ROTAT ( AO,1 )
86. CALL ROTAT (FORCD,1)
87. 4202 ML=MLREF
88. DO 429 J=1,3
89. ML=ML+1
90. IF(JJ(J)-1) 429,428,427
91. 427 XO(ML)=X1(ML)
92. GO TO 426
93. 428 X1(ML)=0.0
94. XO(ML)=0.0
95. 426 V(ML)=0.0
96. A1(ML)=0.0
97. AO(ML)=0.0
98. FORCD(ML)=0.0
99. 429 CONTINUE
100. IF(ANGLE(K,I).EQ.0.)GO TO 430
101. CALL ROTATE(ANGLE(K,I),J4,XO,NDGREE,2)
102. CALL ROTAT( X1,2 )
103. CALL ROTAT( V,2 )
104. CALL ROTAT( A1,2 )
105. CALL ROTAT( AO,2 )
106. CALL ROTAT(FORCD,2 )
107. 430 CONTINUE
108. C COMPUTE NEW DISPLACEMENTS
109. DO 420 I=1,MEQ
110. IF(SMASS(I).EQ.0.) GO TO 420
111. X1(I)=XO(I)+V(I)*DELT+C1*AO(I)+C2*A1(I)
112. 420 CONTINUE
113. CALL FRCIN (NUMEL,NUMNP,NDGREE,IX,E,X1,FINT,OLDS)
114. SUM=0.0
115. DO 425 I=1, MEQ
116. IF(SMASS(I).EQ.0.) GO TO 425
117. A1I=(FORCD(I)-FINT(I))/SMASS(I)

```

```

00007900
00008000
00008100
00008200
00008300
00008400
00008500
00008600
00008700
00008800
00008900
00009000
00009100
00009200
00009300
00009400
00009500
00009600
00009700
00009800
00009900
00010000
00010100
00010200
00010300
00010400
00010500
00010600
00010700
00010800
00010900
00011000
00011100
00011200
00011300
00011400
00011500
00011600
00011700

```

```

118. DA=ABS(A1I+A1(I))
119. IF ( DA .LE. 0.0) GO TO 425
120. SUM=SUM+ABS(A1I-A1(I))/DA
121. A1(I)=A1I
122. CONTINUE
123. IF ( ITERMX .LE. 1 ) GO TO 450
124. CHECK WHETHER TO CONTINUE ITERATION
125. IF(ITER.GE.ITERMX) GO TO 450
126. DA=MEG
127. IF(SUM/DA.GT.EPSI) GO TO 415
128. DO 455 I=1,MEG
129. XO(I)=X1(I)
130. V(I)=V(I)+C3*(A1(I)+A0(I))
131. A0(I)=A1(I)
132. CALL OUTPUT
133. IF(KONTRL(8).LE.0) GO TO 490
134. IF(KONTRL(8).GT.0.AND.MOD(NTSTEP,KONTRL(8)).EQ.0)
135. *CALL RESTRT(1,KONTRL(9),ICLK)
136. CONTINUE
137. IF(NTSTEP.LT.MXSTEP) GO TO 400
138. END OF LOOP
139. RETURN
140. WRITE(6,981) ITERMX,BETA
141. FORMAT(1H0,5X,'DATA ERROR,ITERMX
142. STOP
143. END

```

BETA =1, I10,E15.6)

```

00011800
00011900
00012000
00012100
00012200
00012300
00012400
00012500
00012600
00012700
00012800
00012900
00013000
00013100
00013200
00013300
00013400
00013500
00013600
00013700
00013800
00013900
00014000
00014100
00014200
00014300

```

1.	SUBROUTINE CROSS(A,B,C,BETA,CMAG,IS)	00000100
2.	DIMENSION A(3),B(3),C(3)	00000200
3.		00000300
4.	C PROGRAM FORMS CROSS PRODUCT	00000400
5.	C AND ANGLE BETA. C NORMED TO CMAG	00000500
6.	C = A X B	00000600
7.	C	00000700
8.	C(1)=A(2)*B(3)-A(3)*B(2)	00000800
9.	C(2)=A(3)*B(1)-A(1)*B(3)	00000900
10.	C(3)=A(1)*B(2)-A(2)*B(1)	00001000
11.	AMAG=0.0	00001100
12.	BMAG=0.0	00001200
13.	CMAG=0.0	00001300
14.	DO 10 I=1,3	00001400
15.	AMAG=AMAG+A(I)*A(I)	00001500
16.	BMAG=BMAG+B(I)*B(I)	00001600
17.	CMAG=CMAG+C(I)*C(I)	00001700
18.	AMAG=SQRT(AMAG)	00001800
19.	BMAG=SQRT(BMAG)	00001900
20.	CMAG=SQRT(CMAG)	00002000
21.	BETA=CMAG/(AMAG*BMAG)	00002100
22.	IF (ABS(BETA).GT.1.) BETA=SIGN(1.,BETA)	00002200
23.	BETA=ASIN(BETA)	00002300
24.	IF (IS.EQ.1) RETURN	00002400
25.	DO 20 I=1,3	00002500
26.	C(I)=C(I)/CMAG	00002600
27.	RETURN	00002700
28.	END	00002800

REFERENCES

1. Vail, C. F., "Dynamic Modeling of Automobile Structures from Test Data", Systems Identification of Vibrating Structures, New York, N.Y., 1972
2. Rapin, M. P., "Vehicle Structural Crashworthiness", SAE Paper 700413, May 1970
3. McHenry, R. R. and Miller, P. M., "Automobile Structural Crashworthiness", SAE Paper 700412, May 1970
4. Saczalski, K. J. and Park, K. C., "An Interactive Hybrid Technique for Crashworthy Design of Complex Vehicular Structural Systems", Inter. Conf. on Vehicle Structural Mechanics: A Finite Element Application to Vehicle Design, SAE, Detroit, Mich., March 1974
5. Saczalski, J. J., "Structural Problems Associated with the Prediction of Vehicle Crashworthiness", ASME Applied Mechanics Symposium on Mechanical Problems in Transportation, Detroit, Mich., March 1973
6. Tani, M. and Emori, R. I., "A Study on Automobile Crashworthiness", SAE Paper 700175
7. Kamal, M. M., "Analysis and Simulation of Vehicle to Barrier Impact", SAE Paper 700414
8. Herridge, J. T. and Mitchell, R. K., "Development of a Computer Simulation Program for Collinear Car/Car and Car/Barrier Collisions", Battelle Columbus Laboratories, Report DOT-HS-800-645, January 1972
9. Dynamic Science, "Side Impact Crashworthiness of Full-Size Hardtop Automobiles", Report DOT-HS-800-654, January 1972
10. Selna, L. and Salinas, D., "Dynamic Analysis of Automotive Structural Systems", SAE Paper 700844.
11. Kirioka, K. and Hirata, T., "Computing System for Structural Analysis of Car Bodies", in Advances in Computational Methods in Structural Mechanics and Design, UAH Press, Huntsville, Ala., 1972
12. Kirioka, K., "An Analysis of Body Structures", SAE Trans. 7C, Paper 650026, 1966
13. Kirioka, K.; Bhkubo, Y.; and Hoffa, Y. "An Analysis of Body Structures - Part II" SAE Paper 710157
14. Thompson, J. E., "Control of Structural Collapse in Automotive Side Impact Collisions", PhD Thesis, Univ. of Detroit, Mich., 1972

REFERENCES (Contd)

15. Rong-Chung Shieh, "Basic Research in Crashworthiness II -- Precollapse Dynamic Analysis of Plane, Ideal Elasto-Plastic Frame Structures Including the Case of Collision into a Narrow Rigid Pole Obstacle", Cornell Aeron. Lab., CAL Rept. BV-2987-V-3
16. Young, J. W., "Crash: A Computer Simulator of Nonlinear Transient Response of Structures", Philco-Ford Corp., Palo Alto, Calif., Report DOT-HS-091-1-125-B, March 1972
17. Melosh, R. J., "Car-Barrier Impact Response of a Computer Simulated Mustang", Philco-Ford Corp., Palo Alto, Calif., Report DOT-HS-091-1-125-A, March 1972
18. Krey, R. D. and Key, S. W., "Transient Shell Response by Numerical Time Integration", Int. J Num. Methods in Engr., Vol. 17, 1973
19. Belytschko, T.; Chiapetta, R. L.; and Bartel, H. D.; "Efficient Large Scale Nonlinear Transient Analysis by Finite Elements", to appear Intl. J. for Numerical Methods in Engr.
20. Oden, J. T., Finite Elements of Nonlinear Continua, McGraw-Hill, 1972
21. Argyris, J. H.; Kelsey, S. and Kamel, H.; Matrix Methods of Structural Analysis, B. F. deVeubeke (ed.), AGARDograph 72, Pergamon Press, 1964
22. Wempner, G. A., "Finite Elements, Finite Rotations and Small Strains of Flexible Shells", Intl. J. Solids Struct., Vol. 5, 1969, p 117-153
23. Murray, D. W. and Wilson, E. L., "Finite Element Large Deflection Analysis of Plates", J. Engr. Mech. Div., ASCE, Vol. 95, 1969, p 145-163
24. Belytschko, T. and Hsieh, B. J., "Nonlinear Transient Finite Element Analysis with Convected Coordinates", to be published Intl. J. of Numerical Methods in Engr., 1972
25. Newmark, N. M., "A Method of Computation for Structural Dynamics", J. Engr. Mech. Div., ASCE, Vol. 85, EM3, July 1959
26. Zienkiewicz, O. C. The Finite Element Method in Engineering Science, McGraw-Hill, 1971
27. Meek, J. L., Matrix Structural Analysis, McGraw-Hill, 1971

REFERENCES (Concl)

28. Hartzman, M. and Hutchinson, J. R., "Nonlinear Dynamics of Solids by the Finite Element Method", Computers and Structures - Vol. 2, Pergamon Press, 1972, p 47-77
29. System/360 Scientific Subroutine Package Version III Programmers Manual, IBM Technical Publications Dept., White Plains, N.Y., 1968
30. Archer, R. R. and Lange, C. G., "Nonlinear Dynamic Behavior of Shallow Spherical Shells", AIAA Journal, Vol. 3, No. 12, December 1965
31. Leech, J. W.; Witmer, E. A.; and Pian, T.H.H.; "Numerical Calculation Technique for Large Elastic-Plastic Transient Deformations of Thin Shells", AIAA Journal, Vol. 6, No. 12, December 1968
32. Morino, L.; Leech, J. W.; and Witmer, E. A.; "An Improved Numerical Calculation Technique for Large Elastic-Plastic Transient Deformations of Thin Shells", Part 1 - Background and Theoretical Formulation; Part 2 - Evaluation and Applications, J Appl. Mech., June 1971
33. Shaw, L. M., "Vehicle Front-End Structure Crash Test Program", Monthly Progress Report 16, Contract DOT-HS-046-2-486, Dynamic Science, October 1973

TL 243 .W3
WELCH, R.
FINITE ELE
AUTOMOTIV

Welch

10/6/43

10-19-2011

Form DOT F 172
FORMERLY FORM DC

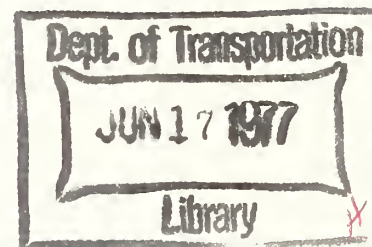
DOT LIBRARY

00092893



TL
243
.W36
v.1

DOT HS-801 846



FINITE ELEMENT ANALYSIS OF AUTOMOTIVE STRUCTURES UNDER CRASH LOADINGS Volume I- Summary

Contract No. DOT-HS-105-3-697

March 1976

Final Report

PREPARED FOR:

U.S. DEPARTMENT OF TRANSPORTATION

NATIONAL HIGHWAY TRAFFIC SAFETY ADMINISTRATION

WASHINGTON, D.C. 20590

Document is available to the public through
the National Technical Information Service,
Springfield, Virginia 22151

The contents of this report reflect the views of IIT Research Institute which is responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policy of the Department of Transportation. This report does not constitute a standard, specification or regulation.

JUN 17 1977

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. DOT HS-801 846	2. Government Accession No.	3. Recipient's Catalog No. Library
4. Title and Subtitle FINITE ELEMENT ANALYSIS OF AUTOMOTIVE STRUCTURES UNDER CRASH LOADINGS Volume I Summary	5. Report Date March 1976	6. Performing Organization Code
7. Author(s) R. E. Welch; R. W. Bruce; T. Belytschko	8. Performing Organization Report No. J6321	10. Work Unit No.
9. Performing Organization Name and Address IIT Research Institute 10 West 35 Street Chicago, Illinois 60616	11. Contract or Grant No. DOT-HS-105-3-697	13. Type of Report and Period Covered Volume I - SUMMARY FINAL REPORT 6-30-1973 6-30-1975
12. Sponsoring Agency Name and Address Department of Transportation, NHTSA 400 7 Street S.W. Washington, D.C. 20590	14. Sponsoring Agency Code	
15. Supplementary Notes		
<p>16. Abstract A research project was undertaken to develop a finite element computer program for use in the dynamic analysis of vehicle structures, including sheet metal, in a crash environment. This research program involved the following major tasks:</p> <ul style="list-style-type: none"> • A technique was developed for the finite element analysis of the dynamic response of plate beam structures involving very large displacements and rotations and elastic-plastic material behavior. The principal feature of this technique involves the decomposition of the element displacement field into rigid body components and deformation components thus allowing the use of a small deflection formulation in the analysis. • A computer program was developed incorporating this analysis procedure for beam and plate elements and rigid links together with appropriate time integration procedures and material property descriptions. 		
17. Key Words finite element computer program, dynamic analysis, vehicles, crashworthiness	18. Distribution Statement Document is available to the public through the National Technical Information Service, Springfield, Virginia 22161	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 21
		22. Price

16. Abstract (Concl)

- A substantial number of test and demonstration problems were analyzed with this computer program ranging from simple classical solutions for beams and plates through large scale simulations of actual crash tests.

PREFACE

This final report, entitled "Finite Element Analysis of Automotive Structures under Crash Loadings" presents the results of a research project undertaken from 1 July 1973 to 30 June 1975 by IIT Research Institute (IITRI) for the Department of Transportation, National Traffic and Safety Administration (NHTSA) under Contract DOT-HS-105-3-697 (IITRI Project J6321). The report is presented in two volumes: Volume I, Summary Final Report, and Volume II, Technical Final Report.

The NHTSA Contract Technical Manager initially was Dr. Michael Chi and subsequently Mr. James Hackney. Dr. R. E. Welch served as the IITRI Project Manager under general supervision of Mr. A. Longinow, Manager, Structural Analysis Section, and Dr. K. E. McKee, Director, Engineering Mechanics. IITRI personnel who made significant technical contributions to the project are R. W. Bruce, M. Smith and R. E. Welch. Prof. T. Belytschko of the Circle Campus, University of Illinois, served as a consultant to the project.

CONTENTS

<u>Section</u>	<u>Page</u>
1. INTRODUCTION	1
2. ANALYSIS	3
2.1 Solution Procedure	3
2.2 Element Formulation	5
2.3 Material Properties	6
3. ANALYSIS RESULTS	9
4. CONCLUSIONS AND RECOMMENDATIONS	19
4.1 Conslusions	19
4.2 Recommendations	20

LIST OF ILLUSTRATIONS

<u>Figure</u>	<u>Page</u>
1. Three-Dimensional Plate Element	7
2. Stub Frame and Sheet Metal Model (Undeformed)	14
3. Stub Frame and Sheet Metal Model (Deformed, 20 msec)	15
4. Stub Frame and Sheet Metal Vehicle Center of Gravity Acceleration	16
5. Stub Frame and Sheet Metal Total Barrier Force	17

1. INTRODUCTION

The increased concern in recent years over vehicle safety has focused, in part, on the ability of a vehicle to sustain a crash event; to absorb, redirect and otherwise manage the severe forces and energy associated with the event; and thereby, to lessen the severity of the environment to which passengers are exposed. In addition to basic vehicle crashworthiness, serious attention has also been devoted to such factors as the propensity of a vehicle to damage in minor crash events and the threat posed to pedestrians or other vehicles by particular vehicle designs.

Since quantitative knowledge of the deformation characteristics of vehicle structures is an intrinsic requirement in all considerations of this type, it is not surprising that the developing concern for vehicle safety has given rise to intense and continuing efforts to provide mathematical descriptions of structural deformations in the crash environment. This report presents the results of a research project undertaken by IIT Research Institute (IITRI) for the Department of Transportation to develop a finite element computer program for use in the dynamic analysis of vehicle structure, including sheet metal, in a crash environment.

Present analytical procedures fall largely into two broad categories which may be termed lumped parameter or equivalent systems and formal approximation techniques (primarily finite element). In the former category, the vehicle is treated as an assemblage of lumped masses and resistance elements which are selected to represent specific subassemblies and components. The properties of the various elements of such models are determined by a static and dynamic crush testing, direct measurement, and the judicious use of simple analysis procedures. Such techniques are valuable in that they provide an efficient and inexpensive method of simulation, are fairly accurate for well established vehicle configurations and, in effect, serve as a depository for previous test and design results. The limitations of this type of procedure are principally that they are heavily reliant on previous testing;

largely limited to well-established vehicle configurations and only weakly tied to basic principles of structural behavior.

The finite element approach represents an attempt to make up for the limitations inherent in the lumped parameter analyses, by employing more formal approximation techniques in the discretization of the structure and a greater reliance on more fundamental structural representations and properties. The limitations of this approach are found in the inherent tendency toward more complicated and expensive computations and the difficulties found in modeling the extensively complex phenomena associated with the crash environment. Such phenomena include large deflections and rotations in the deformed structure, regions of intense curvature (wrinkling), material strain rate effects and interference and contact among structural components during the response. These procedures, indeed, are not totally free of a reliance on testing and analytical judgment and, in fact, can be viewed as a somewhat more formal lumped parameter system and used in connection with the more empirical procedure in hybrid, combination models.

2. ANALYSIS

The response of vehicle structures under crash loadings is a complex process primarily involving:

- transient, dynamic behavior
- complicated framework and shell assemblages
- large deflections and rotations
- extensive plastic deformation

Previous attempts at a formal analysis of this process have been only partly successful due to a variety of limitations which, in particular instances, have included inadequacies in element formulations, material representations or solution procedures. The work presented in this report represents an attempt to develop a finite element program which is specifically tailored to the class of problems inherent in vehicle crash response, and which employs or extends current avenues in finite element analysis which seem best suited to such problems.

In this section major features of the present work are briefly described, and some rationale is offered for their use in the context of vehicle analysis.

2.1 Solution Procedure

The analysis procedure is based on explicit, timewise numerical integration of the equations of motion for the node points (three translations and three rotations per node). The choice of an explicit, rather than implicit, procedure is exploited throughout the analysis by using "lumped" nodal masses and by calculating the internal forces at the nodes from direct integration of the element stress fields without reference to element or assembled stiffness matrices for the structure.

The equations of motion at a typical node point for translations and rotations, respectively, are written as follows:

Translations

$$[M][\ddot{u}_G] = [f_G^E] - [f_G^I] \quad (1)$$

where

- [M] - the diagonal, lumped mass matrix for the node
- $[u_G] = [\ddot{u}_{xG}, \ddot{u}_{yG}, \ddot{u}_{zG}]$ is the acceleration vector for the node referred to a global system $[X_G, Y_G, Z_G]$ common to all nodes
- $[f_G^E]$ - the vector of external, applied loads at the node referred to the global system
- $[f_G^I]$ - the vector of forces at the node in the global system which results from deformations in all elements associated with the node.

Rotation

$$\overline{I_x} \overline{\alpha_x} = \overline{m_x^E} - \overline{m_x^I} + \overline{\omega_y} \overline{\omega_z} (\overline{I_y} - \overline{I_z}) \quad (2a)$$

$$\overline{I_y} \overline{\alpha_y} = \overline{m_y^E} - \overline{m_y^I} + \overline{\omega_z} \overline{\omega_x} (\overline{I_z} - \overline{I_x}) \quad (2b)$$

$$\overline{I_z} \overline{\alpha_z} = \overline{m_z^E} - \overline{m_z^I} + \overline{\omega_x} \overline{\omega_y} (\overline{I_x} - \overline{I_y}) \quad (2c)$$

where all components are referred to a coordinate system $[\bar{x}, \bar{y}, \bar{z}]$ which coincides with the principal axes of the lumped mass moments of inertia at each node and which rotates with the node (i.e., a set of rigid body axes for each node) and where $[\overline{I_x}, \overline{I_y}, \overline{I_z}]$ are the principal mass moments, $[\overline{\alpha_x}, \overline{\alpha_y}, \overline{\alpha_z}]$ are instantaneous angular accelerations, $[\overline{\omega_x}, \overline{\omega_y}, \overline{\omega_z}]$ are the angular velocities, and where $[\overline{m}^E]$ and $[\overline{m}^I]$ are the external and internal moment vectors at the node point.

The orientation of the nodal axes $[\bar{x}, \bar{y}, \bar{z}]$ with respect to the global axes at any time during the motion is established by the components of three unit vectors $\overline{b_1}, \overline{b_2}, \overline{b_3}$ which remain fixed along the nodal axes $[\bar{x}, \bar{y}, \bar{z}]$, respectively, as the node rotates. If the global components of these three unit vectors are

$$\overline{b_1}: (b_{x\bar{x}}, b_{y\bar{x}}, b_{z\bar{x}}) \quad (3a)$$

$$\overline{b_2}: (b_{x\bar{y}}, b_{y\bar{y}}, b_{z\bar{y}}) \quad (3b)$$

$$\overline{b_3}: (b_{x\bar{z}}, b_{y\bar{z}}, b_{z\bar{z}}) \quad (3c)$$

then the components of any vector, \vec{v} , transform from the nodal to global system by the transformation

$$[v_G] = [B][\bar{v}] \quad (4)$$

where the columns of $[B]$ are the components of the three unit vectors.

The choice of an explicit integration procedure and a direct calculation of nodal forces results in a program with minimum (but still substantial) computer storage requirements. This, in turn, is equivalent to a capability of processing reasonably detailed and extensive structural models with relative ease. An early capability for handling relatively large problems is considered essential in developing realistic simulations of actual crash events because of the complex geometry and construction found in real vehicles. Furthermore, this procedure lends itself to a reasonably well organized modular program which admits subsequent change and development, including the further incorporation of an implicit method.

2.2 Element Formulation

The treatment of large displacements and rotations employs a decomposition of the element displacement field into a rigid body rotation and translation associated with a local coordinate system attached to and moving with the element, and a remaining displacement field which describes the deformation of the element relative to the current position of the element axes. This transformation, in effect, removes the average rigid body rotation of the element and allows the use of small or moderate deflection element formulations in the calculation of element forces. In this manner, extremely large rotations and deflections can be accommodated by the analysis with accuracy depending primarily on the size of the elements relative to the curvature of the structure. Although a formal convergence theorem is lacking, the decomposition does hold for infinitesimal regions and numerical studies show excellent agreement with classical solutions. The computer program at present includes low order triangular plate elements and three-dimensional beam elements.

A coordinate system $[\hat{x}, \hat{y}, \hat{z}]$ is embedded in each element and serves to define the rigid body rotation of the element and as a reference for element distortions and forces. The components of a vector, \vec{V} , transform from the element coordinate system to the global coordinate system by the time varying transformation

$$[v_B] = [E][\hat{v}]$$

where the elements of E are

$$\begin{bmatrix} e_{x\hat{x}} & e_{x\hat{y}} & e_{x\hat{z}} \\ e_{y\hat{x}} & e_{y\hat{y}} & e_{y\hat{z}} \\ e_{z\hat{x}} & e_{z\hat{y}} & e_{z\hat{z}} \end{bmatrix}$$

which correspond columnwise with the global components of unit vectors $\vec{e}_1, \vec{e}_2, \vec{e}_3$ oriented with the element axes $[\hat{x}, \hat{y}, \hat{z}]$ respectively. An element coordinate system for a triangular plate in the undeformed and deformed positions is shown in Figure 1. The transformations [E] for each element and [B] for each node together with the components of the normal vector, \vec{n} , at each node or an element, provide sufficient information to calculate deformations in the element. The deformations, in turn, are used to calculate stresses within the element and, subsequently, forces and moments at the nodes. Finally, the nodal forces and moments are entered in the appropriate summations in the equations of motion (equations (1) and (2)).

2.3 Material Properties

The computer program currently uses simple elastic-plastic stress strain laws; a uniaxial relation for beam elements and a biaxial strain hardening Mises model for plates. Element forces and bending moments for given strain fields are calculated by piecewise linear numerical integration of the stresses at selected points in the cross section. The program is designed to accommodate other constitutive relations readily or to accept explicit relations among thrust, moment, extension and curvature. In fact linear relationships are provided at this level which result in more efficient computations for this class of problem.

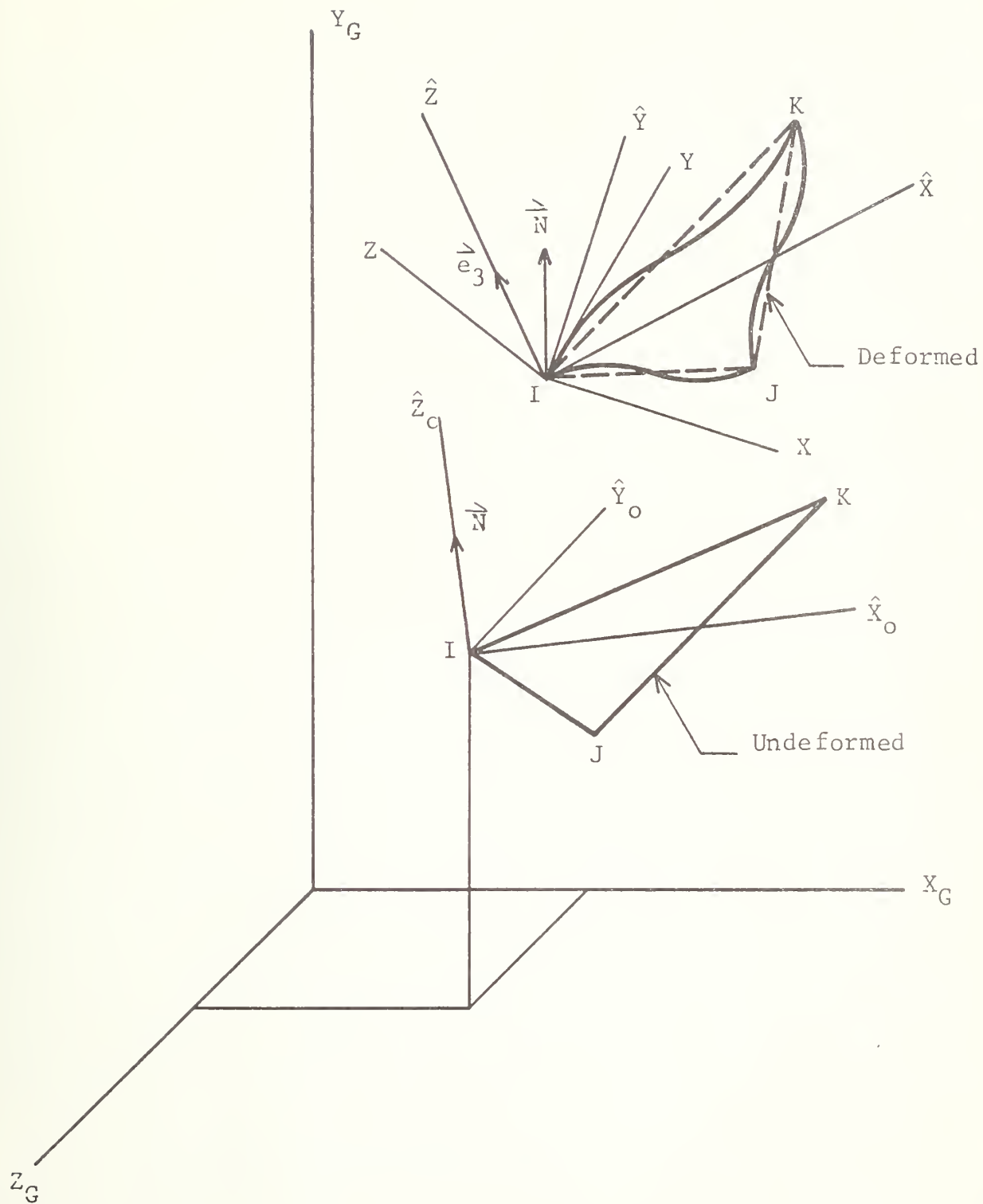


Figure 1 Three-Dimensional Plate Element

3. ANALYSIS RESULTS

A substantial number of test and demonstration problems were treated during the development of the analysis and computer program reported here. For the most part the test problems dealt with relatively simple structural configurations having known results and were selected to isolate and test particular aspects of the formulation and program during development. Thirteen such test problems of successively increasing complexity were formally reported in progress reports. In all cases the analyses were pursued until satisfactory comparison with previous results were obtained or at least to the point at which differences between results could clearly be reconciled on the grounds of differences in loadings, boundary conditions, or ambiguities in the results used for comparison purposes. The contents of these test problems are recapitulated briefly in Table 1.

In addition to the more idealized test problems an effort was made using relatively large scale simulations of actual vehicle crash events. The events which are the objects of these simulations were selected from a series of barrier tests conducted by Dynamic Science which involved frontal impacts at various velocities of a 1968 Plymouth Fury having various portions of the front end structure and machinery removed in different tests. The object of these tests was to isolate and identify the dynamic resistance properties of the various front end components in three groups, namely, frame structure, sheet metal structure and the engine and machinery components. Simulations were developed for two test configurations involving the stub frame alone (test 283-51) and the stub frame and sheet metal together (test 283-53).

The finite element models developed for these two test configurations represent one-half of the structural components forward of the vehicle fire wall and are based on an assumption of symmetry for both the vehicle and its response modes about the center plane.

Table 1

SUMMARY OF TEST PROBLEMS

Problem No.	Description	Element Type and Material Properties	Purpose
1	Cantilever beam with suddenly applied lateral tip load	Two-dimensional beam element; elastic properties	Verify small deflection, dynamic solutions
2	Slightly cambered, simply supported column with sudden axial end shortening	Two-dimensional beam element; elastic properties	Verify large deflection, dynamic solutions
3	Dynamic loading on circular arch	Two-dimensional beam element; elastic properties	Verify dynamic, snap-through solution
4	Impulsive transverse load on axially restrained beam	Two-dimensional beam elements; elastic-plastic properties	Verify elastic-plastic formulation and interaction with axial constraints
5	Plane stress wave propagation	Membrane triangles; elastic and elastic-plastic properties	Verify membrane formulation for in-plane effects
6	Circular membrane with transverse impulsive load	Membrane triangles; elastic and elastic-plastic properties	Verify membrane formulation for out-of-plane effects
7	Elastic, cantilever plate with suddenly applied lateral tip load	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for small deflections
8	Dynamic loading on circular arch	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for plane snap-through problem
9	Elastic cantilever plate	Membrane-hinge line elements, elastic properties	Verify membrane-hinge line formulation

Table 1 (Concl)

Problem No.	Description	Element Type and Material Properties	Purpose
10	Plane-stress wave propagation	Three-dimensional plate bending elements; elastic properties	Verify membrane damping terms
11	Dynamic loading on circular dome	Three-dimensional plate bending elements; elastic properties	Verify plate formulation for three-dimensional snap-through problem
12	Hood structure subjected to end loadings	Three-dimensional plate; elastic-plastic properties	First "representative" vehicle structure and loading
13	Impulsive load on cylindrical panel	Three-dimensional plate; elastic-plastic properties	Verify plate formulation for three-dimensional elastic-plastic response
14	End-on frame crash simulation	Three-dimensional beam elements; elastic-plastic properties	Simulation of actual stub frame test
15	End-on frame and sheet metal crash simulation	Beams and plates; elastic-plastic properties	Simulation of actual frame and sheet metal test

The frame model consists entirely of beam elements and the frame and sheet metal model contains the frame model, plate elements representing the sheet metal, and additional beam elements representing stiffener sections in the hood. In both cases the appropriate inertial properties of the remainder of the vehicle are provided as added masses at the vehicle center of gravity and are attached to the frontal structure by means of rigid links.

As an illustration, we briefly review the results obtained from a simulation of the stub frame and sheet metal barrier test (test 233-53; 44 mph). An isometric view of the finite element mesh for the stub frame and sheet metal model before impact is shown in Figure 2. This model comprises the left front quarter of the 1968 Plymouth employed in the test. The major components of the model are the hood, left front fender, wheel well and frame and radiator support. The wheel well is shown separately for the sake of clarity in the figure but is actually an integral part of the model and is attached along the lines defined by the match points 1 through 4 indicated in the figure. The thickness of all sheet metal elements was taken as 0.039 in. material. The hood portion of the model is free to move relative to the fender at their line of intersection, and, in addition to the plate elements shown, contains 2.0 x 1.0 x 0.05 in. hat section stiffeners at reinforcements. As with the stub frame model additional mass was lumped at the approximate location of the wheel suspension and also at the vehicle center of gravity. In this case all points along the rear edge of the fender and hood as well as the end of the frame were attached to the vehicle center of gravity by means of rigid links. The model was driven by an initial longitudinal velocity of 774 in./sec for all nodes with longitudinal constraints applied at nodes in the forward section of the model successively as they came into contact with the barrier.

The deformed shape of the model at 20 msec after impact is shown in Figure 3. At this point the forward motion of the center of gravity is approximately 15 in. and the forward portions of the

frame assembly and the first two lines of nodes in the hood are indicated in the forward portions of the model with the bumper and bumper brackets collapsed downward and the hood and fender collapsing upward and outward respectively. Separation at the hood and fender junction line is clearly evident and the rear portion of the hood has begun to buckle upward away from the fender. The wheel well is not greatly involved although the forward wall has bent to conform to the distortions in the forward section of the fender.

Longitudinal acceleration at the center of gravity and total barrier force as obtained from test and analysis are shown in Figures 4 and 5. In both cases the test displays the effects of filtering which somewhat obscures the comparisons. The calculated accelerations show an initial peak of about 10 g associated with collapse of the bumper and its supports followed by a larger peak of 30 g as the fender and hood are involved. Maximum acceleration in the test data is about 25 g. The calculated barrier force shows an initial plateau at about 40,000 lbs followed by a sudden rise to about 80,000 lbs as the hood and fender make contact with the barrier. The corresponding maximum barrier load from test (50 percent of full value) is about 60,000 lbs during this phase of the motion.

The test simulations presented here are by no means a complete or exhaustive study even for the limited series of tests which were used for comparison purposes. For a variety of reasons, the results given apply only to the initial phases of the respective crash events and improvements could also be made in the models (the provision of rear wheels or other vertical constraints at the center of gravity, for instance) and in the associated data processing. Despite these limitations, however, the results appear to be a representative portrayal of the phenomena associated with vehicle barrier collisions generally, and a reasonable correlation with the particular test data for which comparisons were attempted.

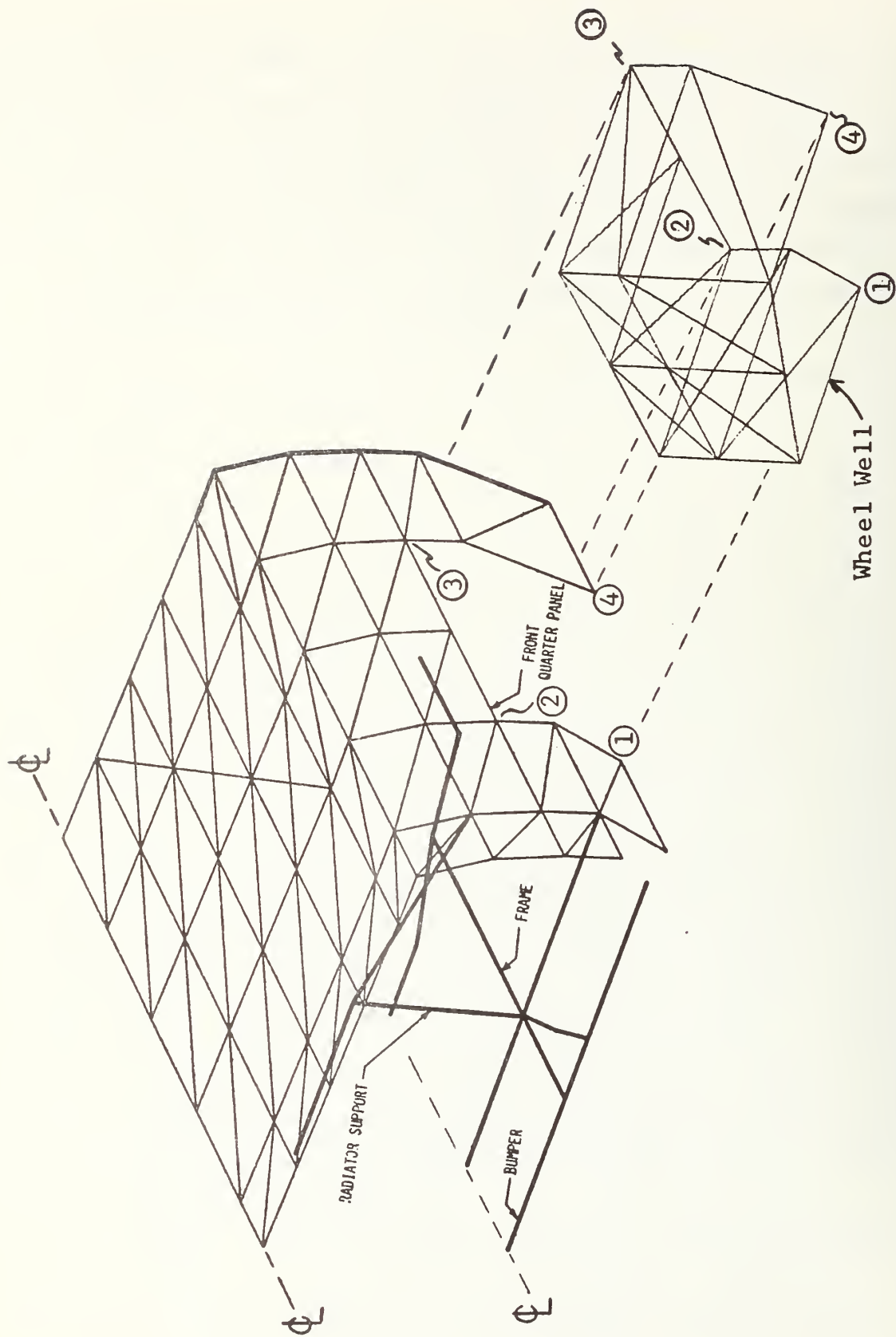


Figure 2 Stub Frame and Sheet Metal Model (Undeformed)

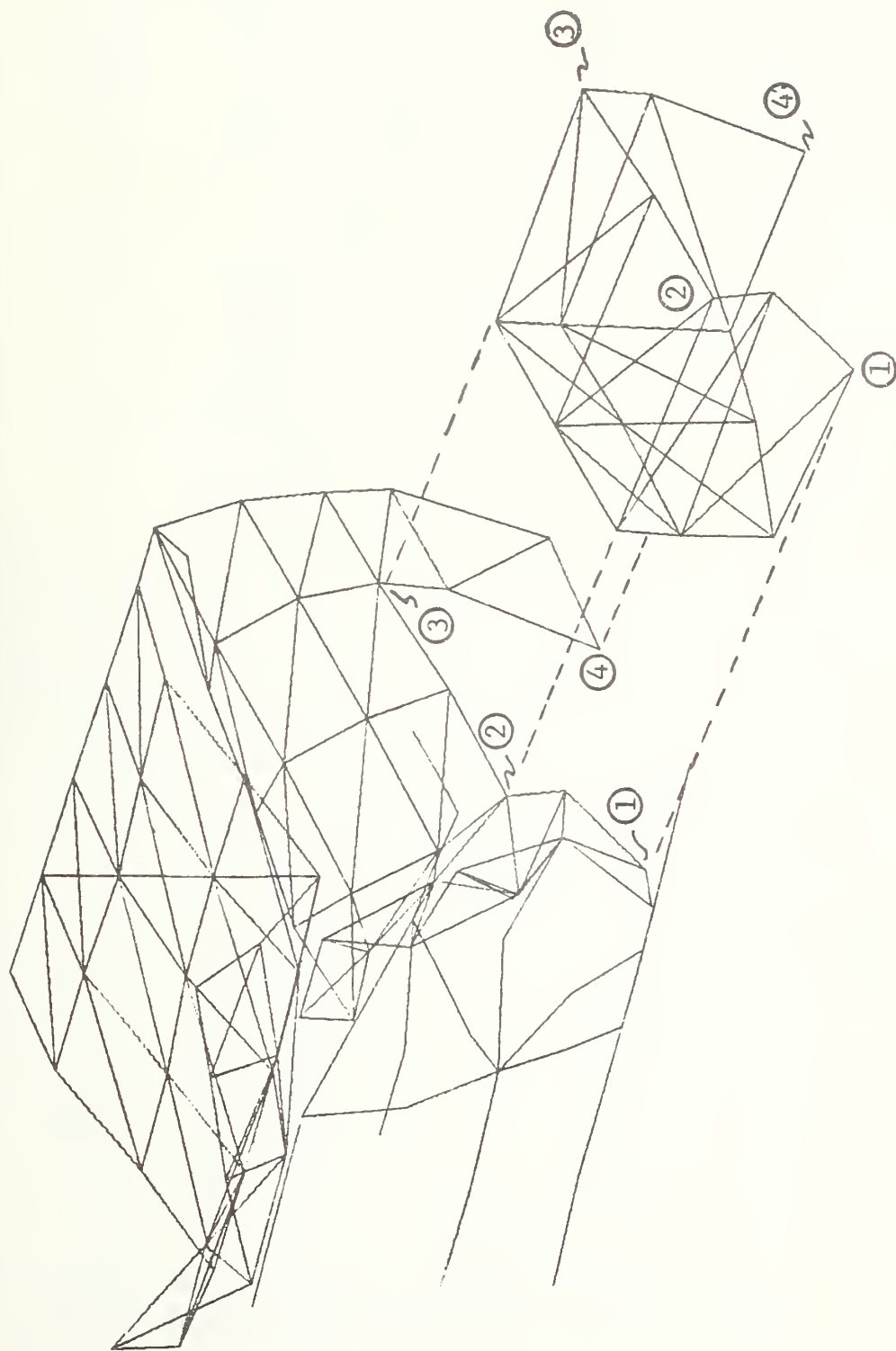


Figure 3 Stub Frame and Sheet Metal Model (Deformed, 20 msec)

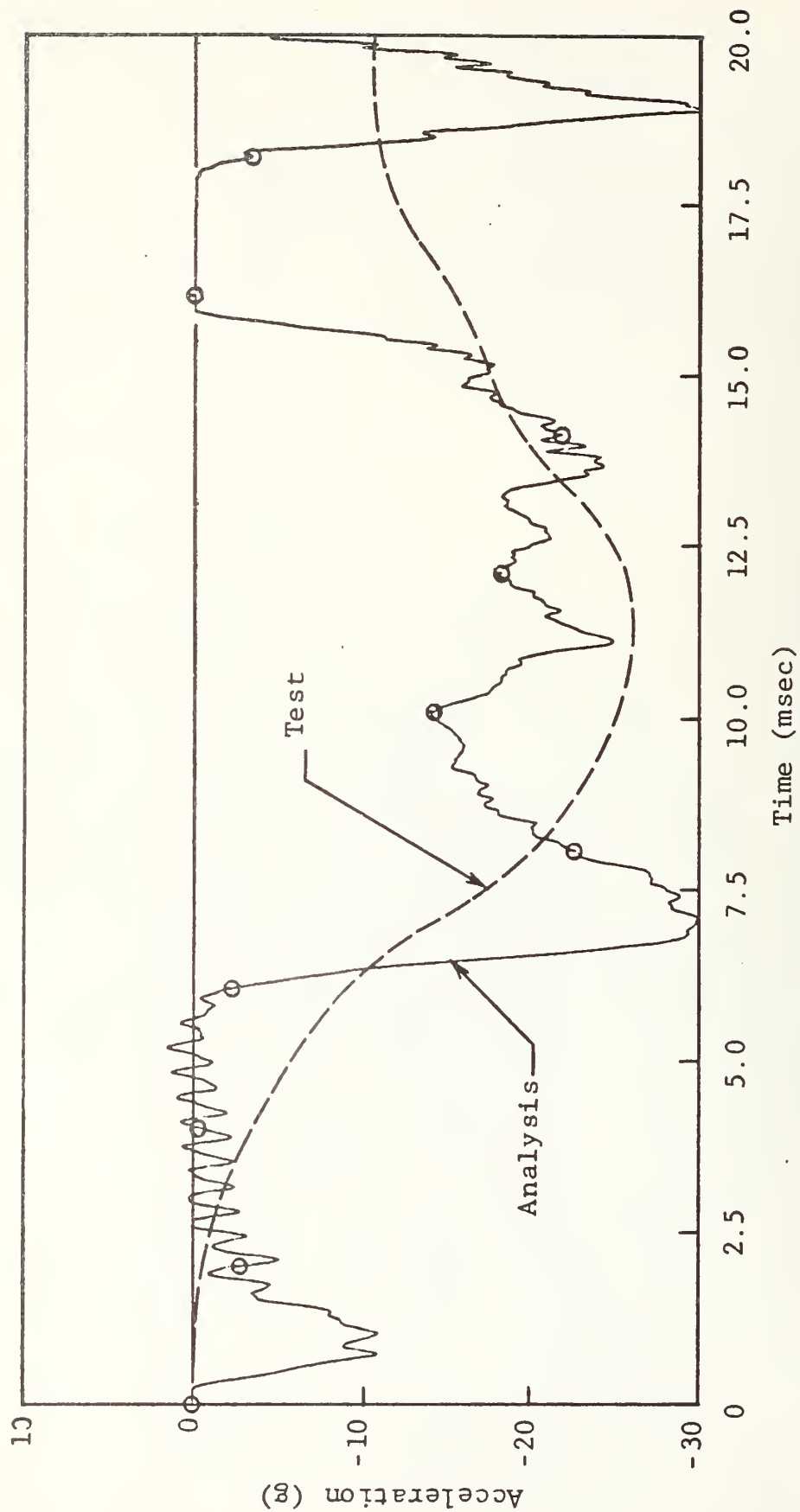


Figure 4 Stub Frame and Sheet Metal Vehicle Center of Gravity Acceleration

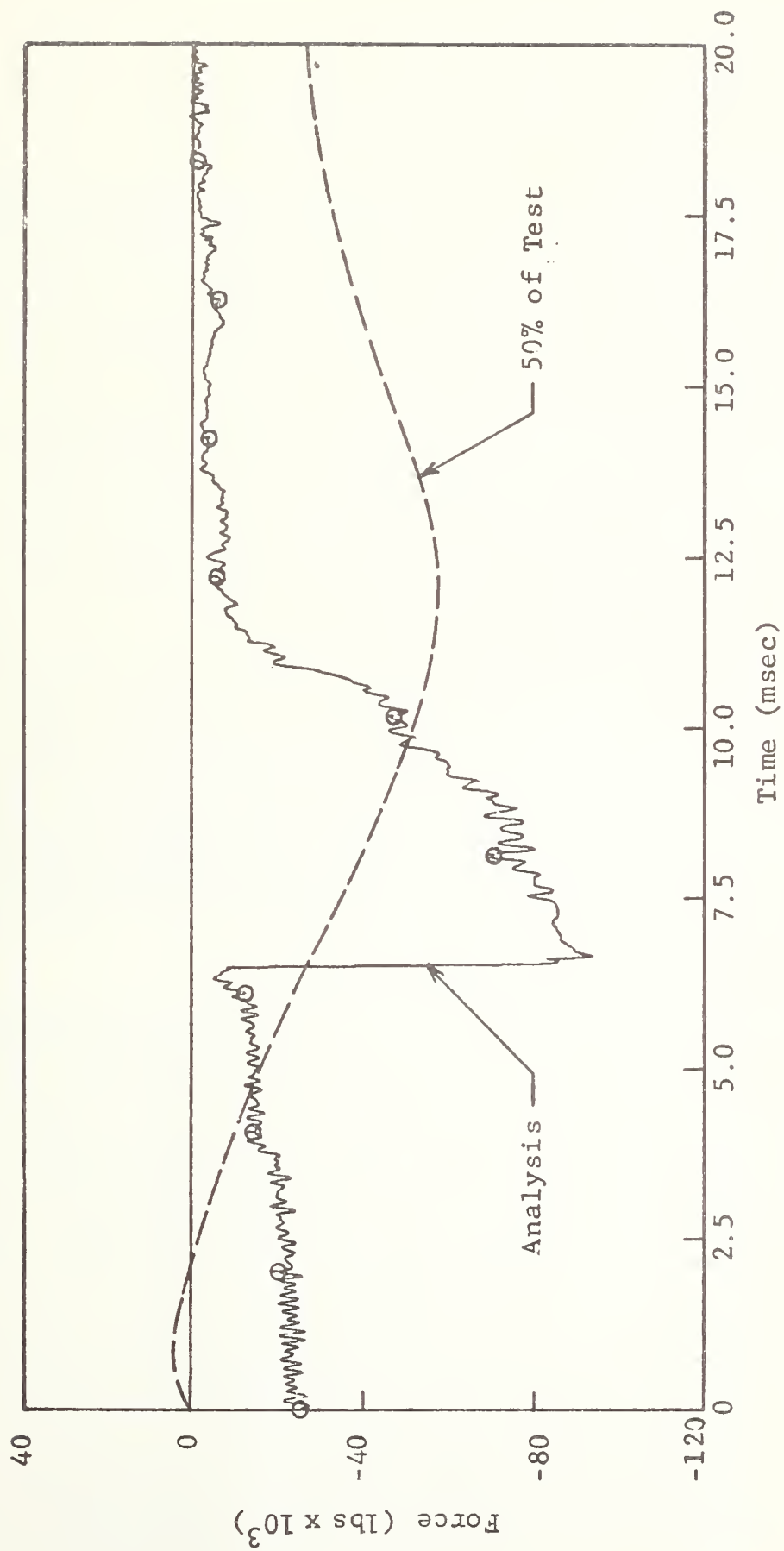


Figure 5 Stub Frame and Sheet Metal Total Barrier Force

4. CONCLUSIONS AND RECOMMENDATIONS

This report presents the results of a research project undertaken by IITRI for DOT to develop a finite element computer program for use in the dynamic analysis of vehicle structures, including sheet metal, in a crash environment. This research program involved the following major tasks:

- A technique was developed for the finite element analysis of the dynamic response of plate beam structures involving very large displacements and rotations and elastic-plastic material behavior. The principal feature of this technique involves the decomposition of the element displacement field into rigid body components and deformation components thus allowing the use of a small deflection element formulation in the analysis.
- A computer program was developed incorporating this analysis procedure for beam and plate elements and rigid links together with appropriate time integration procedures and material property descriptions.
- A substantial number of test and demonstration problems were analyzed with this computer program ranging from simple classical solutions for beams and plates through large scale simulations of actual crash tests.

4.1 Conclusions

Based on the work carried out in this research program our principal conclusions are as follows:

(1) The analysis formulation, while somewhat novel, is a formally sound procedure and a proper and useful approximation technique for the dynamic analysis of beam and plate structures involving large deflections and rotations.

(2) The computer program developed is a correct rendering of this analysis technique and provided accurate results with reasonable efficiency in comparison to available known solutions.

(3) Finally, and most importantly, the computer program is readily applicable to realistic vehicle structures and provides credible results for actual crash events as demonstrated by the simulations of the end-on barrier tests.

4.2 Recommendations

In our opinion the analysis and computer program resulting from this research is a substantial improvement over previous attempts at a finite element analysis of vehicle structures and is a potentially valuable addition to the tools available for the study of vehicle crashworthiness. Although many possibilities suggest themselves in terms of further development and refinement in the analysis procedure and computer program (some of which are noted below) the most important and potentially fruitful activity lies in the immediate and continued application of the analysis to actual vehicles and crash events. Such activity will bring the analysis to real vehicle related problems more quickly and will serve as a guide to its further development and application. Thus our primary recommendation is that provision be made for the continued and extensive use of the program in connection with actual vehicle behavior. Some suggestions for such applications as well as for further analysis and program development are:

Applications

- Additional simulation of controlled crash events along the lines of those attempted in this work but extended to include machinery and drive train components and side-on impacts.
- Use of the program in connection with studies of alternate vehicle designs, including potential structural modifications and configuration changes.
- Use of the program to generate structural crush data for subsequent incorporation in the more simple lumped parameter or hybrid class of programs.
- Comparative studies of the deformation and crashworthiness of existing vehicle designs.

Program Development

- Development of suitable approximate moment-curvature relations for sheet metal as a means of improving computational efficiency.
- Incorporation of plate and beam hinges at selected locations in the main version of the program.

- Development of higher order and special hybrid element formulations possibly to include tearing or wrinkling within the element.
- Development of an implicit rather than explicit integration procedure either as an alternate to the present explicit procedure or for selective use for part of the structure or portions of its response history.

TL 243 .W36 V
WELCH, R. E.
FINITE ELEME
AUTOMOTIVE

Welch

~~14-192949 T~~

Form DOT F 172
FORMERLY FORM DC

